KONVENS 2014

workshop proceedings

October, 8-10 2014

University of Hildesheim,
Germany

The 12th edition of Konvens, the bi-annual German conference is sponsored by the German Society for Computational Linguistics (GSCL), by the special interest group on computational linguistics within the German Linguistics Society (DGfS-CL), and by the Austrian Society for Artificial Intelligence (ÖGAI). We also received support from the German Institute for International Pedagogic Research (Deutsches Institut für Internationale Pädagogische Forschung, DIPF), from the Common Language Resources and Technoloy Infrastructure project CLARIN-D and from the University of Hildesheim.

Universitätsverlag Hildesheim
Marienburger Platz 22
31141 Hildesheim
verlag@uni-hildesheim.de
ISBN 10: 3-934105-47-5
ISBN 13: 978-3-934105-47-8
Hildesheim 2014

This is an electronic publication, it is available via http://www.uni-hildesheim.de/konvens2014

WORKSHOP PROCEEDINGS OF THE 12TH EDITION OF THE KONVENS CONFERENCE

Gertrud Faaß and Josef Ruppenhofer (eds.)

Hildesheim, Germany
October 8 – 10, 2014

Dear Participants of KONVENS 2014,
dear Reader,

it is our pleasure to welcome all attendees of the 12th KONVENS, Konferenz zur Verarbeitung Natürlicher Sprache, and of the co-located workshops in Hildesheim and to make the texts of all contributed papers available to our readership.

Being organized jointly by the German and Austrian community in the field of computational linguistics, as represented by the professional institutions GSCL, Gesellschaft für Sprachtechnologie und Computerlinguistik, ÖGAI, Österreichische Gesellschaft für Artificial Intelligence, and the section on computational linguistics of DGfS, Deutsche Gesellschaft für Sprachwissenschaft, KONVENS has been throughout its history, and continues to be, a privileged forum for the exchange of new ideas, approaches and techniques in the field, bringing together theoretical research, applied work and evaluations.

The 2014 issue of KONVENS is even more a forum for exchange: its main topic is the interaction between Computational Linguistics and Information Science, and the synergies such interaction, cooperation and integrated views can produce. This topic at the crossroads of different research traditions which deal with natural language as a container of knowledge, and with methods to extract and manage knowledge that is linguistically represented is close to the heart of many researchers at the Institut für Informationswissenschaft und Sprachtechnologie of Universität Hildesheim: it has long been one of the institute's research topics, and it has received even more attention over the last few years.

The main conference papers deal with this topic from different points of view, involving flat as well as deep representations, automatic methods targeting annotation and hybrid symbolic and statistical processing, as well as new Machine Learning-based approaches, but also the creation of language resources for both machines and humans, and methods for testing the latter to optimize their human-machine interaction properties. In line with the general topic, KONVENS-2014 focuses on areas of research which involve this cooperation of information science and computational linguistics: for example learning-based approaches, (cross-lingual) Information Retrieval, Sentiment Analysis, paraphrasing or dictionary and corpus creation, management and usability.

The workshops hosted at this iteration of KONVENS also reflect the interaction of, and common themes shared between, Computational Linguistics and Information Science: a focus on on evaluation, represented by shared tasks on Named Entity Recognition (GermEval) and on Sentiment Analysis (GESTALT); a growing interest in the processing of non-canonical text such as that found in social media (NLP4CMC) or patent documents (IPaMin); multi-disciplinary research which combines Information Science, Computer Aided Language Learning, Natural Language Processing, and E-Lexicography with the objective of creating language learning and training systems that provide intelligent feedback based on rich knowledge (ISCALPEL).

As organizers, we are grateful to all contributors and to the invited speakers, Janyce Wiebe, Jacques Savoy, Hinrich Schütze and Benno Stein. We would also like to express our gratitude to all those who lent their time and expertise to the reviewing process, sometimes at short notice. A big thank you is also owed to the organizers of the workshops that KONVENS is hosting this year and to the presenter of Friday's tutorial. Finally, we want to specifically acknowledge all the locals who made the conference and this volume happen: Gertrud Faaß and Josef Ruppenhofer, Fritz Kliche and Stefanie Elbeshausen, Julia Jürgens and Gabriele Irle, and the student assistants Max Billmeier, Melanie Dick, Julian Hocker, Victoria Wandt, and Marie Zollmann.

Christa Womser-Hacker and Ulrich Heid

CONTENTS

NLP4CMC

For a fistful of blogs: Discovery and comparative benchmarking of republishable German content
Adrien Barbaresi, Kay-Michael Würzner 2
Collecting language data of non-public social media profiles
Jennifer-Carmen Frey, Egon W. Stemle, Aivars Glaznieks 11
What does Twitter have to say about ideology?
Sarra Djemili, Julien Longhi, Claudia Marinica, Dimitris Kotzinos and Georges-Elia Sarfati .. 16
Mapping German Tweets to Geographic Regions
Tatjana Scheffler, Johannes Gontrum, Matthias Wegel and Steve Wendler 26
Detecting Irony Patterns in Multi-level Annotated Web Comments
Bianka Trevisan, Melanie Neunerdt, Tim Hemig, Eva-Maria Jakobs and Rudolf Mathar 34
Mining corpora of computer-mediated communication: Analysis of linguistic features in Wikipedia talk pages using Machine Learning methods
Michael Beißwenger, Harald Lüngen, Eliza Margaretha and Christian Pölitz 42
Network of the Day: Aggregating and Visualizing Entity Networks from Online Sources
Darina Benikova, Uli Fahrer, Alexander Gabriel, Manuel Kaufmann, Seid Muhie Yimam, Tatiana von Landesberger and Chris Biemann 48
TWEETDICT Identification of Topically Related Twitter Hashtags
Fabian Dreer, Eduard Saller, Patrick Elsässer and Desislava Zhekova 53
Sentilyzer – A Mashup Application for the Sentiment Analysis of Facebook Pages
Hartmut Glücker, Manuel Burghardt and Christian Wolff 58
Alpes4science project SMS corpus processing and tokenization problems
Eleni Kogkitsidou and Georges Antoniadis 62

ISCALPEL

Bottom up specialized phraseology in CLIL teaching classes
Elisa Corino 68
Towards a collocation writing assistant for learners of Spanish
Margarita Alonso Ramos, Marcos Garcìa Salido and Orsolya Vincze 77
Turning garbage into a writing assistant
Serge Verlinde 89
A lexical database for systematic orthographical teaching and training of German orthography
Hrvoje Hlebec, Wilfried Hehr and Ronny Jauch 95

GermEval

GermEval 2014 Named Entity Recognition Shared Task: Companion Paper
Darina Benikova , Chris Biemann , Max Kisselew , Sebastian Padó 104
Modular Classifier Ensemble Architecture for Named Entity Recognition on Low Resource Systems
Christian Hänig, Stefan Bordag, Stefan Thomas 113

GermEval-2014: Nested Named Entity Recognition with Neural Networks
Nils Reimers, Judith Eckle-Kohler, Carsten Schnober, Jungi Kim, Iryna Gurevych 117
Morphology-aware Split-Tag German NER with Factorie
Peter Schüller 121
HATNER: Nested Named Entitiy Recognition for German
Yulia Bobkova, Andreas Scholz, Tetiana Teplynska, Desislava Zhekova 125
DRIM: Named Entity Recognition for German using Support Vector Machines
Roman Capsamun, Daria Palchik, Iryna Gontar, Marina Sedinkina, Desislava Zhekova 129
BECREATIVE :Annotation of German Named Entities
Fabian Dreer, Eduard Saller, Patrick Elsässer, Ulrike Handelshauser, Desislava Zhekova 134
Nessy: A Hybrid Approach to Named Entity Recognition for German
Martin Hermann, Michael Hochleitner, Sarah Kellner, Simon Preissner, Desislava Zhekova .. 139
Semi-Supervised Neural Networks for Nested Named Entity Recognition
Jinseok Nam 144
Adapting Data Mining for German Named Entity Recognition
Damien Nouvel and Jean-Yves Antoine 149
Named Entity Recognition for German Using Conditional Random Fields and Linguistic Resources
Patrick Watrin, Louis de Viron, Denis Lebailly, Matthieu Constant, Stéphanie Weiser 153
NERU: Named Entity Recognition for German
Daniel Weber and Josef Plötzl 157

GESTALT

IGGSA Shared Tasks on German Sentiment Analysis (GESTALT)
Josef Ruppenhofer , Roman Klinger , Julia Maria Struß , Jonathan Sonntag , Michael Wiegand 164
Saarland University's Participation in the GErman SenTiment AnaLysis shared Task (GESTALT)
Michael Wiegand, Christine Bocionek, Andreas Conrad, Julia Dembowski, Jörn Giesen, Gregor 174
Linn, Lennart Schmeling .....
SentiBA: Lexicon-based Sentiment Analysis on German Product Reviews
Markus Dollmann and Michaela Geierhos 185

NLP4CMC

For a fistful of blogs: Discovery and comparative benchmarking of republishable German content

Adrien Barbaresi
Berlin-Brandenburgische
Akademie der Wissenschaften
barbaresi@bbaw.de

Kay-Michael Würzner
Berlin-Brandenburgische
Akademie der Wissenschaften
wuerzner@bbaw.de

Abstract

We introduce two corpora gathered on the web and related to computer-mediated communication: blog posts and blog comments. In order to build such corpora, we addressed following issues: website discovery and crawling, content extraction constraints, and text quality assessment. The blogs were manually classified as to their license and content type. Our results show that it is possible to find blogs in German under Creative Commons license, and that it is possible to perform text extraction and linguistic annotation efficiently enough to allow for a comparison with more traditional text types such as newspaper corpora and subtitles. The comparison gives insights on distributional properties of the processed web texts on token and type level. For example, quantitative analysis reveals that blog posts are close to written language, while comments are slightly closer to spoken language.

1 Introduction

1.1 Corpora from the web and CMC corpora

Web corpora can be useful to explore text types or genres which are not found in traditional corpora, as well as a whole range of user-generated content and latest language evolutions. The main issues when dealing with such web corpora, be it general-purpose corpora or specific ones, include the discovery of linguistically relevant web documents, the removal of uninteresting parts (or noise), the extraction of text and metadata, and last the republishing of at least part of the content.

So far, there are few projects dealing with computer-mediated communication. In the case of German, the DeRiK project (*Deutsches Referenzkorpus internetbasierte Kommunikation*) features ongoing work with the purpose to build a reference corpus dedicated to computer-mediated communication (Beißwenger et al., 2013).

More specifically, this kind of corpus can used to find relevant examples for lexicography and dictionary building projects, and/or to test linguistic annotation chains for robustness. The DWDS lexicography project at the Berlin-Brandenburg Academy of Sciences already features a good coverage of specific written text genres such as newspaper articles (Geyken, 2007). We wish to conduct further experiments including Internet-based text genres.

1.2 Problems to solve

The problems to solve in order to be able to build reliable computer-mediated communication (CMC) corpora are closely related to the ones encountered when dealing with general web corpora and described above. Specific issues are threefold. First, what is relevant content and where is it to be found? Second, how can information extraction issues be tackled? Last, is it possible to get a reasonable image of the result in terms of text quality and diversity?

Problem 1: Website discovery

First of all, where does one find "German as spoken/written on the web"? Does it even concretely exist or is it rather a continuum? Considering the ongoing shift from web *as* corpus to web *for* corpus, mostly due to an expanding web universe and the potential need for a better text quality, it is obvious that only a small portion of the German web space is to be explored.

Now, it is believed that the plausible distributions of links between hosts follows a power law (Biemann et al., 2013). By way of consequence, one may think of the web graph as a polynuclear structure where the nuclei are quite dense and well-interlinked, with a vast, scattered periphery and probably not so many intermediate pages somewhere in-between. This structure has a tremendous impact on certain crawling strategies. There are ways to analyze these phenomena and to cope with them (Barbaresi, 2014a), the problem being that there are probably different linguistic realities behind link distribution phenomena. While these notions of web science may seem abstract, the centrality and weight of a website could be compared to the difference between the language variant of the public speaker of an organization, and the variants among its basis.

Problem 2: Content extraction

Content extraction is a real problem concerning large web corpora (Schäfer et al., 2013), e.g. because of exotic markup and text genres. While it is generally possible to filter out tag clouds, post lists and left/right columns on webpage scale, the lack of metadata in "one size fits all" web corpora may still undermine the relevance of web texts for linguistic purposes.

In fact, one may argue that decent metadata extraction is necessary for the corpora to become scientific objects, as science needs an agreed scheme for identifying and registering research data (Sampson, 2000).

Problem 3: Text quality

In our particular context, we understand text quality in terms of usefulness for linguistic research. This type of quality has much to do with text integrity, cleaning, and preprocessing, and only addresses to a lesser extent intrinsic factors such as subtlety of language. Our approach deals with opening "black box corpora" and putting them on a test bench.

Undoubtedly, quality of content extraction has an effect on text quality, since the presence of boilerplate (HTML code and superfluous text) or the absence of significant text segments hinder linguistic work. Moreover, there are intrinsic factors speaking against web texts, for instance machine-generated and/or machine-translated content which leads to fluency and grammar correctness problems (Arase and Zhou, 2013), or mixed-language documents (King and Abney, 2013).

In sum, naive approaches to web crawling and web texts may yield positive results when text quantity is more important than text quality, e.g. in machine translation (Smith et al., 2013), but they are bound to impede proper linguistic research. In fact, there are (corpus) linguists who advocate a meticulous selection and extraction of web texts, since size cannot necessarily compensate for lack of quality (Biemann et al., 2013).

Possible ways to address aforementioned problems

We present three possible ways to cope with the issues described in this section. First, design an intelligent crawler targeting specific content types and platforms in order to allow for a fruitful website discovery and, second, to allow for the crafting of special crawling and content extraction tools. Third, find metrics to compare Internet-based resources with already known, established corpora, and assess their suitability for linguistic studies.

2 Retrieval of blog posts and corpus building

2.1 Blog discovery on *wordpress.com*

We chose a specific blogging software, WordPress, and targeted mostly its platform, because this solution compared favorably to other platforms and software in terms of blog number and interoperability. First, *wordpress.com* contains potentially more than 1,350,000 blogs in German. Second, extraction procedures on this website are

https://wordpress.org/
http://wordpress.com/stats

transferable to a whole range of self-hosted websites using WordPress, allowing to reach various blogger profiles thanks to a comparable if not identical content structure.

The crawl of the wordpress.com website has been prepared by regular visits of a tags homepage listing tags frequent used in German posts. Then, a crawl of the tag pages enabled us to collect blog URLs as well as further tags. The whole process has been repeatedly used to find a total of 158,719 blogs.

The main advantage of this methodology is that it takes benefit from the robust architecture of *wordpress.com*, a leading blog platform, as content- and language-filtering are outsourced, which seems to be efficient.

The discrepancy between the advertised and the actual number of blogs can be explained by the lack of incoming links or tags, to a substantial proportion of closed or restricted access blogs, and finally by the relative short crawl of *wordpress.com* with respect to politeness rules used.

2.2 Blog discovery in the wild

A detection phase is needed to be able to observe bloggers "in the wild" without needing to resort to large-scale crawling. In fact, guessing if a website uses WordPress by analysing HTML code is straightforward if nothing was been done to hide it, which is almost always the case. However, downloading even a reasonable number of web pages may take a lot of time. That is why other techniques have to be found to address this issue.

The detection process is twofold, the first filter is URL-based whereas the final selection uses HTTP HEAD requests. The permalinks settings defines five common URL structures for sites powered by WordPress, as well as a vocabulary to write customized ones. A HEAD request fetches the meta-information written in response headers without downloading the actual content, which makes it much faster, but also more resource-friendly, as less than three requests per domain name are sufficient.

Finally, the selection is made using a hard-coded decision tree, and the results are processed using the FLUX-toolchain, Filtering and Language identification for URL Crawling Seeds (Barbaresi, 2013a; Barbaresi, 2013b), which includes obvious spam and non-text documents filtering, redirection checks, collection of host- and markup-based data, HTML code stripping, document validity check, and language identification.

2.3 Content under CC-license

CC-licenses are increasingly popular public copyright licenses that enable the free distribution of an otherwise copyrighted work. A simple way to look for content under CC-licenses resides in scanning for links to the Creative Commons website, which proves to be relatively efficient, and is also used for instance by Lyding et al. (2014). We obtained similar results, with a very good recall and an precision around .65, with can be considered as being acceptable in this context.

That said, as a notable characteristic of internet content republishing resides in the severe copyright restrictions and potential penalties, we think that each and every blog that is scheduled for collection has to be carefully verified, an approach in which we differ from Lyding et al. (2014).

We describe the results of the manual evaluation phase in the evaluation section below. The results of automatic homepage scans on German blogs hosted by *wordpress.com* show that blogs including comments are rather rare, with 12,7% of the total (20,181 websites); 0,8% *at best* under CC license (1,201); and 0,2% *at best* with comments and under CC license (324).

To allow for blog discovery, large URL lists are needed. They were taken out previous web-crawling projects as well as out pages downloaded from *wordpress.com*. We obtained the following yields. There are more than 10e8 URLs URLs from the CommonCrawl project, of which approximately 1500 blogs mostly written in German and potentially under CC-license. The German Wikipedia links to more than 10e6 web documents outside of the Wikimedia websites, in which 300 potential targets were detected. In a list of links shared on social networks containing more than 10e3 different domain names, about 100 interesting ones were found. Last, there were

http://de.wordpress.com/tags/
Such as http://de.wordpress.com/tag/gesellschaft/
http://www.w3.org/Protocols/rfc2616/rfc2616
http://codex.wordpress.org/Using_Permalinks

http://creativecommons.org/licenses/
http://commoncrawl.org

more than 10e6 different URLs in the pages retrieved from *wordpress.com*, in which more than 500 potentially interesting blogs were detected.

In terms of yield, these results show that it is much more efficient to target a popular blog platform. Social networks monitoring is also a good option. Both yield understandably much more blog links than general URL lists. Even if large URL lists can compete with specific search with respect to the number of blogs discovered, they are much more costly to process. This finding consolidates the conclusions of Barbaresi (2014) concerning the relevance of the starting point of a crawl. In short, long crawls have a competitive edge as regards exhaustiveness, but it comes at a price.

The final list of blogs comprises 2727 candidates for license verification, of which 1218 are hosted on *wordpress.com* (45%).

3 Manual assessment of content and licenses

Blog classification has been performed manually using a series of predefined criteria dealing with (1) general classification, (2) content description, and (3) determination of authorship.

First, concerning the general classification, the essential criteria are whether there is really something to see on the page (e.g. no tests such as *lorem ipsum*) and whether it is really a blog. Another classification factor is whether the blog has been created or modified recently (i.e. after 2010-01-01).

Second, concerning the content description, the sine qua nons are to check that the page contains texts, a majority of which being in German, and that the text content is under a CC license. Other points are whether the webpage appears to be spam, whether the content can clearly be classified as dealing with Germany, Switzerland or Austria, whether the content appears to be *Hochdeutsch* or a particular dialect/sociolect, and last if the website targets a particular age group such as kids or young adults.

Third, the authorship criteria are twofold: is the blog a product of paid, professional editing or does it appear to be a hobby; and is the author clearly a woman, a man or a collective?

Concerning the essential criteria, the results of the classification are that 1,766 blogs can be used without restriction (65%), since all the textual content qualifies for archiving, meaning that there is text on the webpage, that it is a blog (it contains posts), that it is mostly written in German and that it is under CC license.

BY-NC-SA	652
BY-NC-ND	532
BY-SA	351
BY	282
BY-NC	129
BY-ND	58

Table 1: Most frequent license types

DE	1497
Unknown	715
AT	146
CH	69
LU	2
NL	2

Table 2: Most frequent countries (ISO code)

The breakdown of license types is shown in table 1, so are the results of country classification in table 2. The CC licensing can be considered to be a sure fact, since theoretically the CC license cannot be overridden once the content has been published. Possible differences between adaptations of the license in the various countries should not be an issue either, because it is done in a quite homogeneous way. The relatively high proportion of BY-NC-ND licenses (30%) is remarkable. While the "-ND" (no derivative works) restriction does not hinder republication as such, its compatibility with corpus building and annotation is unclear, so that such texts ought to be treated with caution.

4 Quantitative evaluation and comparison

4.1 Materials

We present a series of statistical analyses to get a glimpse of the characteristics of the crawled corpora. Content is divided into two different parts, the blog posts (BP), and the blog comments (BC), which do not necessarily share authorship. Due to

the relatively slow download of the whole blogs due to crawling politeness settings, we analyzed a subset of 696 blogs hosted on *wordpress.com* and 280 other WordPress blogs. We cannot calculate how synchronous the subtitles are with the blogs, manual analysis reveals a high proportion of TV series broadcast in the last few years.

Newspaper corpus

The results are compared with established text genres. On one hand, a newspaper corpus which is supposed to represent standard written German, extracted from the weekly newspaper *Die ZEIT*, more precisely the *ZEIT online* section (ZO), which features texts dedicated to online publishing. On the contrary, newspaper articles are easy to date, and we chose to use a subset ranging from 2010 to 2013 inclusive, which roughly matches both size and writing dates of the blogs. There have been digitally generated and are free of detection errors typical for retro-digitized newspaper corpora. ZO is in general considered to be a medium aiming at well-educated people. Therefore, we have picked it as a corpus representing standard educated German.

Subtitle corpus

On the other hand, a subtitle corpus (OS) which is believed to offer a more down-to-earth language sample. The subtitles were retrieved from the OpenSubtitles project, a community-based web platform for the distribution of movie and video game subtitles, then they were preprocessed and quality controlled (Barbaresi, 2014b). Subtitles as linguistic corpora have gained attention by the work of Brysbaert and colleagues (Brysbaert and New, 2009) who showed word frequencies extracted from movie subtitles were superior to frequencies from classical sources in explaining variance in the analysis of reaction times from lexical decision experiments. The reason for this superiority is still somewhat unclear (Brysbaert et al., 2011). It may stem from the fact that subtitles resemble spoken language, while traditional corpora are mainly compiled from written language (Heister and Kliegl, 2012). The analogy between subtitles and spoken language was also the primary motivation to include the OpenSubtitles corpus in the following analyses.

The corpora used in this study are all corpora from the Web. Structural properties of the corpora are shown in table 3. Their sizes are roughly comparable.

4.2 Preprocessing and Annotation

All corpora have been automatically split into tokens and sentences with the help of WASTE, Word and Sentence Tokenization Estimator (Jurish and Würzner, 2013), a statistical tokenizing approach based on a Hidden Markov Model (HMM), using the standard DTiger model. Subsequently, the resulting tokens have been assigned with possible PoS tags and corresponding lemmas by the morphological analysis system TAGH (Geyken and Hanneforth, 2006). The HMM tagger *moot* (Jurish, 2003) has then selected the most probable PoS tag for each token given its sentential context. In cases of multiple lemmas per best tag we chose the one with the lowest edit distance to the original token's surface.

4.3 Analyses

All corpora are aggregated on the level of types, lemmas and annotated types (i.e. type-PoS-lemma triplets) resulting in three different frequency mappings per corpus. Analyses are carried out using the statistical computing environment **R** (R Core Team, 2012).

Quantitative Corpus Properties

Table 3 summarizes a number of standard corpus characteristics. Token and type counts as well as length measures include punctuation. While token length is comparable in all four corpora, sentences in the subtitles are less than half as long as in the other corpora. The proportion of unknown types with respect to the standard-oriented morphological analyzer TAGH is by far smaller in the ZEIT corpus and marginally higher in blog comments than in the other standard-deviating corpora.

Type-Token Ratio

Figure 1 shows the number of types in the four examined corpora as a function of the size of growing corpus samples.

The number of different words with in a corpus is usually interpreted as a measure of its lexical

http://opensubtitles.org

Corpus	Size	⌀ TL	⌀ SL	unkn. T
Token level				
BP	33.0	4.95	20.3	2.76
BC	12.8	4.68	16.0†	2.75
ZO	38.2	5.08	17.5	0.89
OS	67.2	3.90	7.6	1.31
Type level				
BP	1.10	11.3	n/a	24.4
BC	0.56	10.5	n/a	27.3
ZO	0.98	12.2	n/a	13.7
OS	0.83	10.1	n/a	23.9

Size ... Number of tokens (resp. types) in the corpus in millions
TL ... Length of token (resp. type) in characters
SL ... Length of sentences in tokens
unkn. T ... Proportion of tokens (resp. types) unknown to TAGH

† Sentence length was re-computed using a statistical tokenization model (Jurish and Würzner, 2013) trained on the Dortmund Chat Corpus (Beißwenger, 2007). The original value using the standard newspaper model was 22.5, a dubious value.

Table 3: Various properties of the examined corpora.

variance. The plot shows that the OpenSubtitles corpus has a much smaller vocabulary than the three other corpora which are clearly dominated by the blog posts in this respect.

PoS Distribution

Table 4 lists percentage distributions for selected PoS tags on the level of tokens and types. We aggregated some of PoS categories for practical reasons. The figures show that the corpora are rather close in terms of tag distribution with a few remarkable differences. The higher amounts of pronouns and verbs in the subtitles is a direct consequence of shorter sentences. While the proportion of common names drops accordingly, this is not the case for the proper nouns, which validates the hypothesis that the subtitles actually replicate characteristics of spoken language. Besides, the lower proportion of common nouns and higher proportion of proper nouns in the blog comments indicates that it is relevant to study vocabulary diversity.

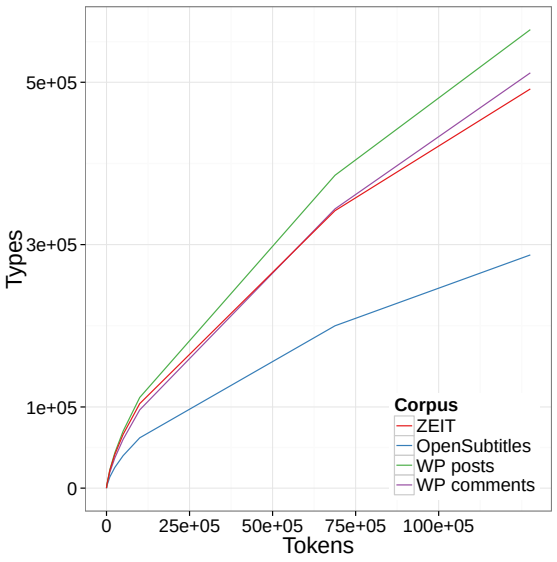


Figure 1: Number of types within random corpus samples (mean, 30 times iterated).

Crps. / PoS	BP	BC	ZO	OS
Content words				
NN	16; 46	13; 42	18; 56	11; 42
NE	3; 22	2; 26	4; 18	3; 27
V*	12; 6	14; 8	13; 6	17; 9
AD*	14; 13	16; 14	13; 14	10; 11
Function words				
ART	8	6	10	5
AP*	8	7	8	4
P*	12	15	12	22
K*	5	5	4	3

Table 4: Percentage distribution of selected PoS (super)tags on token (content and function words) and type level (only content words). PoS tags are taken from the STTS. Aggregation of PoS categories is denoted by a wildcard asterisk. All percentages for function words on the type level are below one percent.

Frequency Correlations

For types shared by all evaluation corpora, Figure 2 shows correlations of their frequencies subdivided by frequency class. Frequency within the OpenSubtitles serves as the reference for frequency class since it is the largest corpus.

Correlations of subtitle frequencies with those from other corpora are clearly weaker than the other correlations while correlations of blog posts

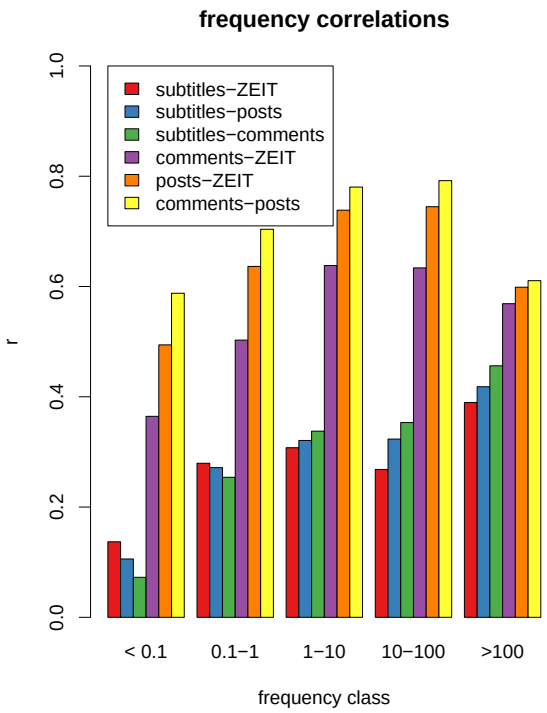


Figure 2: Correlations of type frequencies in different frequency classes.

and comments are always higher. The general pattern is the same in all frequency classes but the differences between the single correlation values are smaller in the highest and lowest range.

Vocabulary Overlap

Figure 3 shows overlaps in the vocabulary of the four corpora using a proportional Venn diagram (Venn, 1880). It has been generated using the *Vennerable* (Swinton, 2009) **R** package which features proportional Venn diagrams for up to nine sets using the Chow-Ruskey algorithm (Chow and Ruskey, 2004). The diagram is arranged into four levels each corresponding to the number of corpora sharing a type. The yellow layer contains types which are unique to a certain corpus. Types shared by two corpora are mapped to light orange levels while dark orange levels contain types shared by three corpora. Types present in all four corpora constitute the central red zone. The coloring of the borders of the planes denotes the involved corpora. In order to abstract from the different size of the data sets involved and to allow for an intuitive comparison of the proportions within the diagram, we included only the 100,000 most frequent words from each evaluation corpus into the analysis.

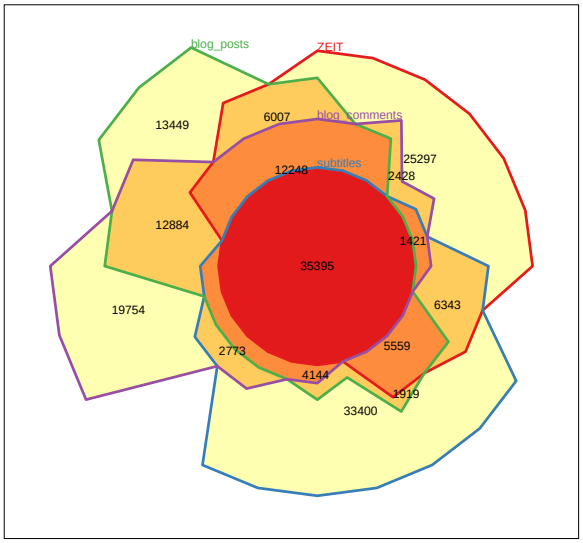


Figure 3: Venn diagram for the 100,000 most frequent words from each evaluation corpus.

Despite the heterogeneous nature of the corpora, there is a large overlap of roughly a third of the types between the four samples (red plane). Each sample contains a significant amount of exclusive tokens. The overlap between blog posts and comments is by far the largest on the second level while the one between blog posts and subtitles is the smallest. There is also a surprisingly large overlap between blog posts, comments and the ZEIT.

4.4 Discussion

The analyses above show large differences between the OpenSubtitles corpus on one and the ZEIT corpus on the other hand. These differences concern sentence length with much shorter sentences in the OS corpus; the amount of unknown words which includes non-standard word forms and (less frequent) named entities; frequency correlations which shows large frequency deviations in the medium frequency range and PoS distributions with fewer nouns and more verbs for the subtitles. We interpret these results as resembling some of the differences between spoken and written language.

In almost all analyses, blog content is found to be closer to the ZEIT corpus then to the OpenSubtitles corpus. This might be expected for the posts but it is somewhat surprising concerning the comments which are to a great extent discourse-like communication. Nonetheless, our quantitative results are in accordance with qualitative results on that matter (Storrer, 2001; Dürscheid, 2003).

In exception to that pattern, the amount of tokens unknown to TAGH in the blog samples is comparable to the value for the OpenSubtitles. This is caused by phenomena such as typos, standard-deviating orthography and *netslang* frequently observed in computer-mediated text and communication. In order to guarantee reliable linguistic annotation of blog posts and comments, emphasis will have to be put on improving existing and developing specific methods for automatic linguistic analysis.

5 Conclusion

First of all, our results show that it is possible to find blogs in German under Creative Commons license. The crawling and extraction tools seem to give a reasonable image of blog language, despite the fact that the CC license restriction impedes exploration in partly unknown ways and probably induces sociological biases.

We introduced evidence to try to classify blog corpora. Post content and comments seem to be different in nature, so that there is a real interest in separate analysis, all the more since it is possible to perform text extraction and linguistic annotation efficiently enough to allow for a comparison with more traditional or established text types. In this regard, a corpus comparison gives insights on distributional properties of the processed web texts.

Despite the presence of atypical word forms, tokens and annotation UFOs, most probably caused by language patterns typically found on the Internet, token-based analysis of blog posts and comments seems to bring these corpora closer to existing written language corpora.

More specifically, out-of-vocabulary tokens with respect to the morphological analysis are slightly more frequent in blog comments than in the other studied corpora. Concerning the lexical variance, blog posts dominate clearly, even if the higher proportion of proper nouns in the blog comments signalizes a promising richness regarding linguistic studies. Vocabulary overlap is best between blog posts and comments. However, a slight difference subsists between them, the latter being potentially closer to subtitles, as the PoS tag distribution seems to corroborate the hypothesis that subtitles are close to spoken language.

We believe that the visualizations presented in this article can help to answer everyday questions regarding corpus adjustments as well as more general research questions such as the delimitation of web genres.

Future work includes updates of the resources as well as full downloads of further blogs. Longer crawls as well as tries on other blog platforms might be a productive way to build bigger and potentially more diverse transmissible corpora. Additionally, more detailed annotation steps could allow for a thorough interpretation.

Part of the processing toolchain used in the experiments is available online under an open-source license. The corpora mentioned in this paper are available upon request.

Acknowledgments

Blog classification has been performed by Sophie Arana. Bryan Jurish has helped with the fine-tuning of the linguistic processing chain.

References

Yuki Arase and Ming Zhou. 2013. Machine Translation Detection from Monolingual Web-Text. In *Proceedings of the 51th Annual Meeting of the ACL*, pages 1597–1607.

Adrien Barbaresi. 2013a. Challenges in web corpus construction for low-resource languages in a post-BootCaT world. In Zygmunt Vetulani and Hans Uszkoreit, editors, *Proceedings of the 6th Language & Technology Conference, Less Resourced Languages special track*, pages 69–73.

Adrien Barbaresi. 2013b. Crawling microblogging services to gather language-classified URLs. Workflow and case study. In *Proceedings of the 51th Annual Meeting of the ACL, Student Research Workshop*, pages 9–15.

Adrien Barbaresi. 2014a. Finding Viable Seed URLs for Web Corpora: A Scouting Approach and Comparative Study of Available Sources. In Roland

https://github.com/adbar

Schäfer and Felix Bildhauer, editors, *Proceedings of the 9th Web as Corpus Workshop*, pages 1–8.

Adrien Barbaresi. 2014b. Language-classified Open Subtitles (LACLOS): download, extraction, and quality assessment. Technical report, BBAW. https://purl.org/corpus/german-subtitles.

Michael Beißwenger, Maria Ermakova, Alexander Geyken, Lothar Lemnitzer, and Angelika Storrer. 2013. DeRiK: A German reference corpus of computer-mediated communication. *Literary and Linguistic Computing*, 28(4):531–537.

Michael Beißwenger. 2007. Corpora zur computervermittelten (internetbasierten) Kommunikation. *Zeitschrift für germanistische Linguistik*, 35(3):496–503.

Chris Biemann, Felix Bildhauer, Stefan Evert, Dirk Goldhahn, Uwe Quasthoff, Roland Schäfer, Johannes Simon, Leonard Swiezinski, and Torsten Zesch. 2013. Scalable Construction of High-Quality Web Corpora. *Journal for Language Technology and Computational Linguistics*, pages 23–59.

Marc Brysbaert and Boris New. 2009. Moving beyond Kučera and Francis: A critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for American English. *Behavior Research Methods*, 41(4):977–990.

Marc Brysbaert, Matthias Buchmeier, Markus Conrad, Arthur M Jacobs, Jens Bölte, and Andrea Böhl. 2011. The word frequency effect: A review of recent developments and implications for the choice of frequency estimates in German. *Experimental Psychology*, 58(5):412–424.

Stirling Chow and Frank Ruskey. 2004. Drawing area-proportional venn and euler diagrams. In Giuseppe Liotta, editor, *Graph Drawing*, volume 2912 of *Lecture Notes in Computer Science*, pages 466–477. Springer.

Christa Dürscheid. 2003. Medienkommunikation im Kontinuum von Mündlichkeit und Schriftlichkeit. Theoretische und empirische Probleme. *Zeitschrift für Angewandte Linguistik*, 38:35–54.

Alexander Geyken and Thomas Hanneforth. 2006. TAGH: A Complete Morphology for German based on Weighted Finite State Automata. In *Finite State Methods and Natural Language Processing*, volume 4002 of *Lecture Notes in Computer Science*, pages 55–66. Springer.

Alexander Geyken. 2007. The DWDS corpus: A reference corpus for the German language of the 20th century. In Christiane Fellbaum, editor, *Collocations and Idioms: Linguistic, lexicographic, and computational aspects*, pages 23–41. Continuum Press.

Julian Heister and Reinhold Kliegl. 2012. Comparing word frequencies from different German text corpora. In Kay-Michael Würzner and Edmund Pohl, editors, *Lexical Resources in Psycholinguistic Research*, pages 27–44. Potsdam Cognitive Science Series. vol.3.

Bryan Jurish and Kay-Michael Würzner. 2013. Word and Sentence Tokenization with Hidden Markov Models. *JLCL*, 28(2):61–83.

Bryan Jurish. 2003. A Hybrid Approach to Part-of-Speech Tagging. Final report, Kollokationen im Wörterbuch, Berlin-Brandenburgische Akademie der Wissenschaften.

Ben King and Steven Abney. 2013. Labeling the Languages of Words in Mixed-Language Documents using Weakly Supervised Methods. In *Proceedings of NAACL-HLT*, pages 1110–1119.

R Core Team, 2012. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. http://www.R-project.org/.

Geoffrey Sampson. 2000. The role of taxonomy in language engineering. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 358(1769):1339–1355.

Roland Schäfer, Adrien Barbaresi, and Felix Bildhauer. 2013. The Good, the Bad, and the Hazy: Design Decisions in Web Corpus Construction. In Stefan Evert, Egon Stemle, and Paul Rayson, editors, *Proceedings of the 8th Web as Corpus Workshop*, pages 7–15.

Jason R. Smith, Herve Saint-Amand, Magdalena Plamada, Philipp Koehn, Chris Callison-Burch, and Adam Lopez. 2013. Dirt Cheap Web-Scale Parallel Text from the Common Crawl. In *Proceedings of the 51th Annual Meeting of the ACL*, pages 1374–1383.

Angelika Storrer. 2001. Getippte Gespräche oder dialogische Texte? Zur kommunikationstheoretischen Einordnung der Chat-Kommunikation. In Andrea Lehr, Matthias Kammerer, Klaus-Peter Konerding, Angelika Storrer, Caja Thimm, and Werner Wolski, editors, *Sprache im Alltag. Beiträge zu neuen Perspektiven in der Linguistik*, pages 439–466. De Gruyter.

Jonathan Swinton. 2009. Vennerable. http://r-forge.r-project.org/projects/vennerable.

John Venn. 1880. On the diagrammatic and mechanical representation of propositions and reasonings. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 10(59):1–18.

Collecting language data of non-public social media profiles

Jennifer-Carmen Frey, Egon W. Stemle, Aivars Glaznieks
Institute for Specialised Communication and Multilingualism
European Academy of Bozen/Bolzano, Viale Druso 1, Italy
{jennifer.frey,egon.stemle,aivars.glaznieks}@eurac.edu

Abstract

In this paper, we propose an integrated web strategy for mixed sociolinguistic research methodologies in the context of social media corpora. After stating the particular challenges for building corpora of private, non-public computer-mediated communication, we will present our solution to these problems: a Facebook web application for the acquisition of such data and the corresponding meta data. Finally, we will discuss positive and negative implications for this method. [1]

1 Introduction

The exploration of new genres of computer-mediated communication (CMC) has most recently become one of the central research objectives when creating and analysing CMC corpora. Most research projects focus on publicly available language data. For example, there is a lot of research on data such as wikipedia articles and corresponding discussion sites (e.g. Storrer, 2012), public chats (e.g. Beißwenger and Storrer, 2012), twitter statuses (e.g. Greenhow and Gleason, 2012), and public social networking profiles (e.g. Pérez-Sabater, 2012). So far, the attention paid to private conversation in CMC research has been sparse [2], resulting in an under-representation of authentic private communication settings in the current picture of social media language.

The small number of corpora of private CMC may result from various difficulties related to data acquisition. Compared to publicly available data, the acquisition of private data is considerably more difficult in terms of privacy issues[3], technical implementation and sampled data retrieval. Obtaining private CMC data is time-consuming for both the researchers and the participants because direct interaction between the two is needed. Additionally, the data acquisition process may involve various media breaks, this in turn would cause problems in terms of consistency of data transfer and would increase the risk of possible data loss. Consequently, the whole process may turn into a rather expensive endeavour.

However, new forms of data acquisition could help to handle the emerging constraints. Therefore, we developed a method, using technical solutions that rose out of the current settings of media usage, for the acquisition of linguistically relevant social media content. After providing an overview of the underlying research project (Section 2) and listing the most urgent challenges when dealing with individual and user-based data of non-public social media profiles (Section 3), we present our fully integrated web solution, implemented as a Facebook web application (Section 4). Finally, in order to emphasize the relevance of our approach, we discuss its advantages

[2] But see for example the Swiss SMS Corpus http://www.sms4science.uzh.ch

[3] Albeit, thoroughly considering the recommendations on internet research by Markham and Buchanan (2012), for instance, can be exhausting enough.

and disadvantages (Section 5).

2 The DiDi Project

The DiDi project investigates the characteristics of South Tyrolean language use on the Social Networking Service (SNS) Facebook by following a sociolinguistic user-based perspective on language data (Androutsopoulos, 2013). Therefore, the goal is to create a corpus of individual SNS communication that can be linked to other user-based data such as age, web experience and communication habits. We gathered sociodemographic information through an online questionnaire and collected the language data of the entire range of social interactions, i.e. publicly accessible data as well as non-public conversations (status updates and comments with restricted privacy settings, private messages, and chat conversations meaning instant messaging) written and published just for friends or a limited audience.[4] Two month after the release of the app, we ended the data acquisition phase with about 150 users that interacted with the app, offering access to their language data and answering the questionnaire. From those we collected 21.400 private messages, 9.248 status updates (6.784/73% non-public) and 5.399 wall comments (4.622/86% non-public), that matched our specific research criteria (L1 German, living in South Tyrol, texts originated in 2013).

3 Challenges for the Acquisition of non-Public SNS Data for CMC Corpora

Bolander and Locher (2014) and Beißwenger and Storrer (2008) discuss, among others, general issues and challenges for corpora of publicly available CMC data. When dealing with non-public data the stated issues of data acquisition for CMC corpora become more demanding: *legal concerns* add to *ethical issues* already mentioned in previous research, and *technical demands* related to *authentic* data retrieval and the linking of *mixed resources* (i.e. language data and sociolinguistic meta information) get more challenging.

For technical and legal reasons of data acquisition interaction between the user and the researcher becomes an inevitable necessity. Whereas the *legal* situation of the research usage of user-generated language data is still under debate for generally public data, the trend leans towards seeking user consent. User-generated language data is always bound to copyright restrictions therefore making every modification, (re)publication or citation, potentially problematic (cf. Baron et al., 2012). Furthermore, ethical considerations researchers should also respect when doing data acquisition of private personal data, demand that such a consent is to be received in advance and that the user data is anonymised (Beißwenger and Storrer, 2008). For non-public data, this legal and ethical issues are of course even more critical.

But also *technical constraints* make it necessary to interact with the user, to gain access to the data. Most media platforms therefore offer interfaces for third parties to obtain access via an explicit permission from the user. With regard to this, a user consent for the usage of private data is legally – and often technically – necessary.

Finding a *representative sample of participants* for the corpus is another problem that, in fact, many corpus creation projects face. Often expensive public relation campaigns and incentives are necessary to get users to participate in projects where the requested data is personal, often intimate and not written for the public. There are different approaches in gathering the otherwise non-accessible private data, most of them asking for individual submissions of language data by the users as for example in the recent "What's up Switzerland?" project[5]. There, participants of the project need to register and send single threads of conversation via mail, following detailed submission guidelines.

As we wanted to make the participation process as attractive as possible, we tried to find another way to gather the data: Particularly, as we considered this to be tedious for users and researchers, and also troublesome because of privacy doubts on the user side and authenticity doubts on the research side. Speaking of non-public language data, the users might feel that

[4] For a detailed description of the project cf. Glaznieks and Stemle (Submitted).

[5] `http://www.whatsup-switzerland.ch/en/`

their writing does not reflect "proper" language use, and hence brush it up before donating it. Such modifications however reduce the *authenticity* of the data and should be avoided when analysing the language use in social media.

For the reasons of gaining user consent and sociolinguistic meta-data with the highest privacy for participants (i.e. no personal interaction, no backtracking via mail addresses, etc.) and collecting authentic language data, automatic data collection should be preferred over submission by users. Besides it will make the participation more attractive by simplifying the procedure of sharing language and meta data in an integrated, time-saving and genuine way (i.e. the participation stays within the same platform, using the platform's interfaces and methods that are already familiar for users).

4 Non-Public SNS Data for CMC Corpora – the DiDi Web App

To address the challenges described in section 3, we designed a Facebook web application that manages all the necessary interaction with the participants.[6] A complete run-through consists of the following steps:

1. informing potential participants about the research project, the privacy policy and the data usage declaration;
2. providing options for the user to choose which content to share (private inbox and/or personal wall) and thereby increasing the transparency for the user about which data will actually be retrieved;
3. authenticating the user via the Facebook login dialogue (by using the Facebook API);
4. obtaining the consent to use, save and republish the user's data (via the web application as well as via the Facebook infrastructure for privacy policies);
5. managing the registered user and the granted permissions via the Facebook login dialogue and the Facebook API;
6. requesting an anonymous and individual user identifier for the survey client, saving permission flags, and enlisting the user into an internal database;
7. redirecting to the survey for the acquisition of the user's meta information;
8. providing dynamic feedback to the user about the current progress of the project (e.g. the amount of participants);
9. providing the possibility to share the application with Facebook friends to attract more users.

[6]The source code of the DiDi web application is available at `https://bitbucket.org/commul/didi` for the main application and at `https://bitbucket.org/commul/didi-ws` for the corresponding web service.

5 Properties of an App-Supported Data Acquisition

An app-supported data acquisition has advantageous properties but also some constraints that should be considered.

5.1 Advantages

The most important advantage is that the application facilitates the access to authentic, unrevised and non-public domains of every-day computer-mediated communication. The data is received in a well-defined format and is genuinely machine-readable, easy to restructure or to join with other (social networking) content. Basic annotations, concerning, for instance creation time, privacy settings of content, links to multi-modal elements or devices used for text production, already come with the data.

With respect to the participation process, the web application keeps it as slim and simple as possible. It takes users solely two clicks to donate their language data. After this, the user will be redirected to an integrated online questionnaire. For logging in and accepting the terms of privacy of the app, users do not need to register anywhere but will simply follow the familiar Facebook routines for apps. There is no one-to-one interaction between an authenticated person and a researcher as this would raise privacy issues and doubts in the consistency of anonymisation. Furthermore, legal and ethical constraints are met within the online setting without additional effort. Meta information of the questionnaire and actual language data are automatically linked with an anonymous user identifier, provided by Facebook individually

for every registered user of the app. Therefore, the identifiers can be used even with third-party survey services without privacy problems.

Moreover, the app procedure facilitates the isolation of user acquisition and interaction with the actual crawling of language data. The application only manages registered users. After logging in, the application grants access to the user's account for a period of 60 days. Thus, using such a web application enables efficient data crawling. While users do not have to wait for the language data download to complete, the risk of data loss and other loading and saving issues decreases, as data can be retrieved in independent processes whenever performance and memory capacities allow it best. Furthermore, server or system failures do not result in data loss since the data can be requested repeatedly.

Finally, there are various possibilities to support the attractiveness of the research project. Dynamic feedback can be given through the application surface allowing participants to be part of a collective community project. The application can be easily shared as Facebook post, blog comment, twitter status, e-mail or any other media content. After having finished the survey, participants can directly share the application with their friends via Facebook. This workflow is genuine to social media contexts and addresses interested users wherever they happen to be. In addition, participants can be reached by Facebook via targeted advertising campaigns that address a specific user subset and are usually paid by conversions or actual reach of the advertisement.

5.2 Demands and problems of the application strategy

Using such a web application may save a lot of manual work in data acquisition and be inevitably necessary for the data accessibility. However, it raises the demands on design, development and hosting of the application. Therefore, it increases human workload, required expertise and technical demands. For example, an appropriate infrastructure is needed first of all for the setup of the application (webserver, system and server reliability and monitoring, timely response in case of failures). Secondly, the appropriate infrastructure is needed for a secure and safe data transmission and storage (internal server storage and services, encrypted data transmission and connection, etc.) to ensure anonymity and protect the users' privacy.

In addition to the implementation of the general app functionality and its technical requirements, usability concerns and graphical interface design principles should also be considered to make the software engaging and easy to handle. Therefore, to minimize the efforts in expertise and workload a general app infrastructure for obtaining facebook and/or other social media content as a reusable module for different projects could be a future objective in CMC corpus research.

Another problem within the app approach is the remaining chance of data loss. Within our application design it was not obvious for the users that the data crawling does not happen at the actual moment of participation. The disassociation of these two procedures favours a comfortable participation and crawling procedure, but may also lead to false presumptions. Users may disauthorise the application directly after the participation and hence avert the subsequent data crawling unintentionally. In addition, Facebook is able to refuse data requests even with valid permissions if they suspect the application to be malware. This could occur when downloading a lot of data or when users repeatedly mark the application as untrustworthy. So, there is no guarantee for a complete access to the data during the entire permission period. Thus, the project's ethical and reliable behaviour should be clear and comprehensible.

6 Conclusion

The proposed web app strategy for the acquisition of SNS data facilitates the collection of non-public language data that would otherwise be very complicated or even unfeasible. Therefore, we take our app as a step towards a general and reusable infrastructure that might help to keep the technical efforts for further development low and hence help people to profit from the advantages of this approach.

References

Jannis Androutsopoulos. 2013. Online Data Collection. In Christine Mallinson, Becky Childs, and Gerard Van Herk, editors, *Data Collection in Sociolinguistics: Methods and Applications*, chapter 14, pages 236–250. Routledge, New York.

Alistair Baron, Paul Rayson, Phil Greenwood, James Walkerdine, and Awais Rashid. 2012. Children Online: A survey of child language and CMC corpora. *International Journal of Corpus Linguistics*, 17(4):443–481.

Michael Beißwenger and Angelika Storrer. 2008. Corpora of Computer-Mediated Communication. In Anke Lüdeling and Merja Kytö, editors, *Corpus Linguistics. An International Handbook.*, Handbücher zur Sprach- und Kommunikationswissenschaft, chapter 17, pages 292–308. De Gruyter, Berlin.

Michael Beißwenger and Angelika Storrer. 2012. Interaktionsorientertes Schreiben und interaktive Lesespiele in der Chat-Kommunikation. *Zeitschrift für Literaturwissenschaft und Linguistik: Lili*, 42(168):92–125.

Brook Bolander and Miriam A Locher. 2014. Doing sociolinguistic research on computer-mediated data: A review of four methodological issues. *Discourse, Context & Media*, 3:14–26.

Aivars Glaznieks and Egon W. Stemle. Submitted. Challenges of building a CMC corpus for analyzing writer's style by age: The DiDi project. *JLCL*.

Christine Greenhow and Benjamin Gleason. 2012. Twitteracy: Tweeting as a new literacy practice. In *The Educational Forum*, volume 76, pages 464–478. Taylor & Francis.

Annette Markham and Elizabeth Buchanan. 2012. Ethical Decision-Making and Internet Research: Recommendations from the AoIR Ethics Working Committee (Version 2.0). Technical report, AoIR Ethics Working Committee, December.

Carmen Pérez-Sabater. 2012. The linguistics of social networking: A study of writing conventions on facebook. *Linguistik online*, 56(6/12):81–93.

Angelika Storrer. 2012. Neue Text- und Schreibformen im Internet: Das Beispiel Wikipedia. In Juliane Köster and Helmuth Feilke, editors, *Textkompetenzen in der Sekundarstufe II*, pages 277–306. Fillibach, Freiburg.

What does Twitter have to say about ideology?

Sarra Djemili
ETIS - UCP/ENSEA/CNRS 8051
UCP, Cergy-Pontoise, France
sarahdjemili@yahoo.fr

Julien Longhi
CRTF - UCP/EA 1392
UCP, Cergy-Pontoise, France
Julien.Longhi@u-cergy.fr

Claudia Marinica
ETIS - UCP/ENSEA/CNRS 8051
UCP, Cergy-Pontoise, France
Claudia.Marinica@u-cergy.fr

Dimitris Kotzinos
ETIS - UCP/ENSEA/CNRS 8051
UCP, Cergy-Pontoise, France
Dimitrios.Kotzinos@u-cergy.fr

Georges-Elia Sarfati
STIH/ EA 4509
Paris Sorbonne, France
georgesarfati@gmail.com

Abstract

Political debates bearing ideological references exist for long in our society; the last few years though the explosion of the use of the internet and the social media as communication means have boosted the production of ideological texts to unprecedented levels. This creates the need for automated processing of the text if we are interested in understanding the ideological references it contains. In this work, we propose a set of linguistic rules based on certain criteria that identify a text as bearing ideology. We codify and implement these rules as part of a Natural Language Processing System that we also present. We evaluate the system by using it to identify if ideology exists in tweets published by French politicians and discuss its performance.

1 Introduction

Political and ideological debates have been a part of our political and societal functions for many years, to some extend since the first steps of the civilization. One could argue that the opinions of others are important to us in order to make for example a responsible decision regarding the electability of a particular candidate, to look beyond appearances and be able to judge the character of people. This includes evaluating their intelligence and leadership abilities, but it also involves learning about people's stance on various issues. On the other hand, fewer people have anymore the time and will to put the effort to go through the analysis of short or longer texts that position people and opinions or even worse sometime even reading them does not provide adequate answers. Moreover, the explosion of the internet brought multiple ways of communicating one's political opinions, thus making the whole process more difficult. In this context, microblogging services like the Twitter network give people the ability to express themselves with brevity but with speed and with less preparation thus exposing them more easily into the public. So, identifying or even studying ideology has become an even more challenging task (Riabinin, 2009).

Apart from that, studying ideology has always been a main issue in French discourse analysis domain. However, a semantic analysis of ideology has not been fully and rigorously developed (see Rastier 's assessment in (Rastier, 2011)), so even nowadays, these analyses lack of scientific description and especially rigorous evaluation. In that respect, one of the objectives of this article is to provide rigorous criteria for the identification of ideologies in tweets but also to implement them in a tool which allows their identification and validation. The complementarity with research in

computer science provides answers to longstanding questions in the literature of discourse analysis. The choice of working on Twitter is justified by the fact that it is characterized as a new genre of political discourse as we showed in (Longhi, 2013), and due to its brevity it reflects a semantic condensation possibly to be favorable to ideologies. The work presented here is evaluated over text (tweets) that are in French, which was an obvious choice given the fact that the authors live and work in France and that we draw the rules we propose from criteria suggested for text in French. Apparently similar approaches could exist in other languages; transferring though either the criteria or the rules or both does not seem to work given the particularities in each language and the fact that our work is based on expressing and quantifying linguistic rules.

Political discourses were already analyzed in the literature, but this area is still young especially when the object of research is text produced in social media environments and when additionally we aim to identify relevant tweets based on the existence of ideological references in them. Some existing studies focus on discovering political affiliations in informal web-based contents like news articles (Zhou et al., 2011), political speeches (Dahllf, 2012) and web documents (Durant and Smith, 2007; Durant and Smith, 2006; Efron, 2006). Political data-sets such as debates and tweets are explored for classifying users' positions (Walker et al., 2012; Somasundaran and Wiebe, 2010) and also for predicting election results (O'Connor et al., 2010) or the political party affiliation (Conover et al., 2011). These works use for prediction the content and other corpus specific properties such as hashtags, social networks, etc. Other works use ideological political beliefs for party prediction (Gottipati et al., 2013) exploiting likewise specific text properties.

Concerning ideology detection, existing works are based on simple linguistic models as in (Gerrish and Blei, 2011) where the authors predict the voting behavior of legislators on the basis of bag-of-words representations from the proposed bills and deduct legislators' political tendencies. Another type of works use annotated corpus in order to infer lexical characteristics of the ideology; one of these works is (Sim et al., 2013) where authors have used an HMM model (Hiden Marcov Model) to deduct ideologies in candidate discourse during the campaign cycle of united-states in 2012. Similarly, in (Iyyer et al., 2014) the authors introduce a model for political ideology detection using a recursive neural network (RNN) in order to detect ideological influence at sentence level. The authors state that the resulting model can correctly identify ideological influence in complex syntactic constructions.

The ideology was defined by multiple authors in multiple occasions. According to Erikson and Tedin in (2003), the ideology is a "...set of beliefs about the proper order of society...". Knight (2006) points out the fact that "Specific ideologies crystallize and communicate the many beliefs, opinions and values of an identifiable group...". This definition is basic, limited to the political camp (right, left, etc.). The ideology refers obviously to the "content" of a discourse, but it can also rely on the "form"; in this context, the discourse analysis field proposes valuable criteria to identify ideology.

In this work, we propose a set of rules that can be used to identify ideology in tweets and other short text messages. These rules stem from Sarfati's work (2014) on the necessary criteria to classify text as bearing any kind of ideology. On top of that we implemented these rules as part of a Natural Language Processing System that allows its use over the large corpuses that can be collected e.g. from Twitter. We evaluated these rules using actual tweets from French politicians.

This paper is structured as follows: in the next section we present Sarfati's criteria and we describe the steps taken to transform them to linguistic rules. Then we describe how we implement these rules as part of a Natural Language Processing (NLP) System which we detail more in the beginning of the section (section 3). In section 4 we evaluate the implemented rules over a carefully validated corpus of tweets and present our preliminary results and first conclusions. We conclude the paper in section 5 by providing a sum up of the work so far and some pointers for future research.

2 From Sarfati's criteria to linguistic rules

The main objective of this paper is to detect whether or not a tweet is an ideology tweet, but not to classify it further according to the ideological references it carries. The work introduced by Sarfati (2014) provides the definition of the necessary criteria for a text to be classified positively as an ideology bearing text. Our effort is to transform the proposed criteria into linguistic rules and implement them as part of a Natural Language Processing System. Sarfati describes seven criteria on ideology: some of them are used just to characterize the type of the ideology or to describe it generally, but others are more definitive, permitting to detect ideology in text. Thus, in this study we concentrate on the five criteria presented below; a tweet is ideological if and only if it satisfies all five criteria and all the criteria have the same weight.

- Criterion 1: the deictic scope of the ideology is the one of a discourse state pretending to erase any clutch mechanism, any dependence on an enunciation place or any spatiotemporal context. The ideological discursive state claims *timelessness*;
- Criterion 2: the level of heterogeneity of the ideology consists in the negation itself of the mixed discourse, since under its strategic claim of transparency (universality) and of timelessness (transhistorical), ideology is structured as a *homogeneous* discourse, discursively smooth;
- Criterion 3: the ideology aims to produce the illusion of *timelessness* and it states an effective relevance for all times;
- Criterion 4: the reflexiveness level of the ideology consists in the fact of not pretending referring only to itself, that is to say that the ideology is its own end;
- Criterion 5: the ideology is *polychronous* as it pretends grouping all the temporal perspectives and canceling them.

Below we describe the (linguistic) rules that correspond/implement to each one of the seven criteria. These rules fall within the framework of the theory of discursive objects, developed by Longhi in (2008) for the concept of discursive object and in (2014) for the theory itself. One goal of this theory is to assign formal markers to discursive operations, in order to provide discourse analysis from pragmatic and declarative criteria. More generally, the theory of discursive objects opens up Sarfati's theory to linguistic corpora.

Criterion 1 is implemented by:

Rule 1: no spatiotemporal deixis marks, such as: here (*ici* - fr), there (*là-bas* - fr), now (*maintenant* - fr), tomorrow (*demain* - fr), etc.

Rule 2: no interlocution subjects, such as: I (*je* - fr), you (*tu*, *vous* - fr), we (*nous* - fr), and occurrence of non-subjects, such as: he/she (*il/elle* - fr).

Rule 3: no proper nouns specifying places, people or factual data that are too precise.

Criterion 2 is implemented by:

Rule 4: in order to validate the universality and the homogeneity characteristics, no modalization marks should occur, such as: to seem to (*sembler* - fr), to appear (*paraître* -), to be able to (*pouvoir* - fr), to have to (*devoir* - fr). These marks outline speaker's attitude towards the statement. Moreover, this rule is confirmed also by the absence of punctuation marks such as "?" and "!" outside of a reported speech.

Rule 5: reduce the argumentation: no argumentative connectors, such as: but (*mais* - fr), so (*donc* - fr), because (*parce que*, *puisque* - fr), etc.), or neutral connectors, such as: and (*et* - fr), moreover (*de plus* - fr), etc.

Criterion 3 is implemented by:

Rule 6: for timelessness, the verb should be at present tense stating out a general truth. The past and future tenses should be present less frequently.

Criterion 4 is implemented by:

Rule 7: referring only to itself, the ideology should not contain other discourse marks, such as: double quotes, according to (*selon* - fr), as X says/thinks (*comme X dit/pense* - fr), etc.

Criterion 5 is implemented by:

Rule 6 is adequate in order to validate this criterion.

Since a tweet is identified as ideological if and only if it satisfies all the criteria, then, conse-

quently, a tweet has to satisfy all seven rules described above in order to be identified as ideological.

3 Integrating linguistic rules in Natural Language Processing tools

The rules described in the previous section will allow us to determine if a tweet is ideological or not. In order to develop a system implementing these rules, we evaluate the possibility of integrating the linguistic rules into existing tools of Natural Language Processing (NLP).

Moreover, the implementation of these rules in our system requires a morpho-syntactic analysis in order to determine the part-of-speech category for each word in a tweet: verb, adjective, noun, preposition, etc. For this purpose, we also need to use a suite of NLP tools that carries the corresponding functionality. Thus we reviewed the available open source[2] NLP APIs that we will detail in the following subsection.

3.1 Morpho-syntactic analysis in NLPs

Part-of-speech (POS) tagging is one of the most fundamental parts of the linguistic analysis, a basic form of syntactic analysis which has important applications in NLP. The goal of this study is to analyze the POS tagging APIs available for French language and to compare them in order to evaluate their capabilities and limits, and to finally select one or more of them to use. In our study, we are searching for the following elements: verb tenses, adjectives and nouns objective or subjective, personal pronouns, connectors, proper nouns, space and time markers. We tested and evaluated three well-known POS taggers:

- Stanford POS Tagger[3]: offers a Java implementation of the log-linear POS tagger provided by the Stanford NLP group. The provided library allows the user to tag words in the text. The tagger has to load a trained file (named model) containing the necessary information for the tagger. Several trained models are provided by Stanford NLP group for different languages, including French; for French, the model is based on the prelabeled French corpus named Treebank.

- Apache Open NLP[4]: the Apache Open NLP library is a machine learning based toolkit for natural language text processing. It supports the most common NLP tasks, such as tokenization, sentence segmentation, POS tagging, chunking, etc. These tasks are usually required to build more advanced text processing services. The French model is also based on Treebank corpus.

- Wikimeta[5]: is a labeling tool based on NLGbAse content. NLGbAse is a system producing metadata and components for natural language processing, semantic analysis, and labeling tasks. NLGbAse transforms encyclopedic text contents into structured knowledge according to the Linked Data and the Semantic Web principles. NLGbAse metadata are used to produce resources and training corpora for information extraction tools like Wikimeta. Wikimeta detects named entities, and links them to their RDF description available as Linked Data. The semantic labeling web service API provides a REST-compliant, unique access point for all text-mining and content analysis functionality. The French Java API of Wikimeta also provides TreeTagger, a POS Tagger, and a frequency analysis tool.

In order to compare the POS taggers presented above, we test the performance of their APIs on a set of 100 tweets representing 1920 words. To this end, each API annotates the tweets' words with the corresponding tags, and then we manually compare the results and compute the error rate for each API. The results, presented in Table 1, point out (1) that, regarding the error rate, the Wikimeta Tagger outperforms the other taggers, and (2) that Wikimeta proposes a larger number of tags.

Moreover, the analysis allowed us to determine that, on the one hand, Stanford POS Tagger makes no distinction between nouns and proper

[2]We surveyed only open source APIs both because they are open to anyone to use and the code is available to extend as needed

[3]http://nlp.stanford.edu/software/tagger.shtml

[4]https://opennlp.apache.org/

[5]http://www.wikimeta.fr/

	Stanford POS Tagger	Apache Open NLP Tagger	Wikimeta Tagger
Error rate	$2,5\%$	$2,55\%$	$2,39\%$
Number of tags	8	13	37

Table 1: Comparison of the results provided by Stanford POS, Apache Open NLP and Wikimeta Taggers.

nouns, between verbs and past participles, and does not tag accordingly verbs' tenses, articles and amounts. On the other hand, Apache Open NLP Tagger does not detect punctuation marks and, as Stanford POS Tagger, does not detect verbs' tenses, articles and amounts although it offers more details than the later.

To conclude, Wikimeta allows us to detect all the elements that we need in order to implement the linguistic rules, such as: verbs' tenses, connectors, proper nouns, personal pronouns. Moreover, it is able to give details concerning proper names, and distinguish between places and people through the detection of named entities (it connects named entities to their RDF description from the linked data).

Based on the results detailed above, we decided to use Wikimeta's API to develop our system for detecting ideological tweets.

3.2 Integration of rules

In this section, we detail how we integrate, using Wikimeta, in our system, the linguistic rules that we created starting from Sarfati's criteria in section 2, and which technical issues this development introduces.

Rule 1: In order to implement this rule, we use initially Wikimeta to analyze the tweet as it provides three interesting tags: NTIME, NDAY and NMON which detect temporal entities. Then, given that we are interested in seventeen (17) spatio-temporal markers, we create a set with all these markers and check if they appear in a tweet. For example, now (*maintenant* - fr), tomorrow (*demain* - fr), etc.

Rule 2: Equally, for interlocution subjects, using Wikimeta we can easily check if the tweet's text contains: I (*je* - fr), you (*tu*, *vous* - fr), we (*nous* - fr), me (*moi* - fr), etc.

Rule 3: For this rule, Wikimeta can spot all proper nouns existing in the tweet. Since proper nouns can be represented by abbreviations, Wikimeta can also help since it detects abbreviations and labels them with the "ABR" tag.

Rule 4: To check if a tweet contains one of the four modal verbs, we first need to find the infinitive form of the verbs in the tweet. To do that, we use a second API[6] that ensures the lemmatization; this API was developed by the Natural Language Processing group of Sheffield University. Thus, we can compare the returned verb with the four (4) ones in our list. Concerning the question (?) and exclamation (!) marks, we just check if they exist in the tweet.

Rule 5: Concerning the use of connectors, we look for the argumentative ones referring to a pre-existing list.

Rule 6: For rule 6, we use Wikimeta in order to detect the tense of each verb in the tweet. But, since a text can contain at the same time verbs at different tenses, we have to compute the most dominant verb tense in the tweet. To this end, we count the occurrence of each verb tense in the tweet by using three classes corresponding to past, present and future tenses.

Rule 7: Detecting discourse markers in French language was addressed by several works such as (Poulard et al., 2008; Giguet and Lucas, 2001; Buvet, 2012; Mourad and Desclés, 2003). The automatic identification of citations is not an obvious task as the identification of marks of reported speech, especially in the indirect case, is based on combinatorial heterogeneous linguistic units (Buvet, 2012). Authors proposed in (Giguet and Lucas, 2001) a syntactic strategy that we exploit. It consists of locating three unknown elements: the source (of the citation - speaker), the reported speech and the text introducing the reported speech (e.g.: declared that (*a déclaré* -fr)). They used phrase-oriented criteria as computing indices: typographical signs (punctuation, capitalization), and morpho-syntactic and position-based elements for computing a three-value variable: source, reported speech and the introduc-

[6] http://staffwww.dcs.shef.ac.uk/people/A.Aker/activityNLPProjects.html

tory text. For that, they established a model for French corpus admitting two designs, according to the two different types of speech - direct or indirect - detailed in the following:

- the first one is a direct speech with the form X explained that... (*X a expliqué que...* - fr);
- the second one is a indirect speech with the form ...explained X (*...a expliqué X* - fr).

Moreover, for the direct speech, the double quotation mark outlines the opening of reported speech and the end of a reported speech (words in double quotes " "). For the indirect speech, he (*il* - fr) points out the presence of a speaker and that (*que* - fr) marks that a indirect reported speech might follow.

In tweets' context, detecting direct speech is equivalent to identifying mentions having reply type (tweets that started with a @username) in addition to double quote signs. We also check the verbal speaker expressions. For indirect speech, markers like the ones mentioned above are identified. Additionally, we used the table given in (Mourad and Desclés, 2003) containing statistics about the most used verbs for detecting the speaker.

3.3 System operation

In order to apply the previous linguistic rules on a significant number of tweets, we developed the system presented in Figure 1.

The system takes as input a set of political tweets and provides as result the set of the ideological tweets. A morpho-syntactic analysis is done on the tweets by Wikimeta API allowing POS annotation and detection of named entities. A tweet is identified by the system as ideological only if it satisfies *all* of the seven linguistic rules presented above, knowing that all the rules have the same weight in the system. For each tweet the system notes the rules that it satisfies.

4 Application to Twitter Dataset

4.1 Tweets

In recent years, social media activity has reached unprecedented levels. Hundreds of millions of users now participate in online social networks and forums, subscribe to microblogging services or maintain web diaries (blogs). Twitter is currently the major microblogging service, with more than 255 million monthly active users who send more than 500 million Tweets (short text messages of up to 140 characters) per day[7]. They use tweets to report their current thoughts and actions, comment on breaking news and even engage in discussions.

4.2 Corpus Description

Nowadays, political tweets are considered by linguistic researchers as a new form of political discourse (Longhi, 2013). Through their tweets, politicians aim to make public their (new) ideas and convictions, but, also to convince the voters that their (the politicians') goals, expectations and actions are the ones to follow and support. In this context, we propose to test our system on a political tweets corpus as there is a bigger probability to contain ideological texts. Moreover doing this, we expect to reduce noise as politicians usually use more standard French when tweeting, avoiding much of web-slang.

The corpus of tweets that we used in our experiments was established by (Longhi et al., 2014) to serve two research projects: the "CoMeRe" project which aims to establish a set of corpus-mediated communications networks, and the "Digital Humanities and Data Journalism" project which aims to develop interdisciplinary research collaborations allowing to analyze political corpus produced via new ways of communication. The corpus was built starting from seven (7) French politicians of six (6) political parties. In order to generate political tweets, we started from a set of lists citing these politicians (7087 lists), and we selected those lists that have tweeted at least 6 times and which description contains the word *politics* - 120 lists remaining. Finally, 2934 tweets were recovered.

In order to be sure that we select politicians' tweets (and not for example ones from journalists), we worked by keeping only the accounts cited in more than 12 lists; we have finally 205 politicians who were tweeting. For these 205 accounts we got the last 200 tweets of each on 27 March 2014 (34,273 tweets). This allows us to have a corpus focusing on the period between the

[7] https://about.twitter.com/company

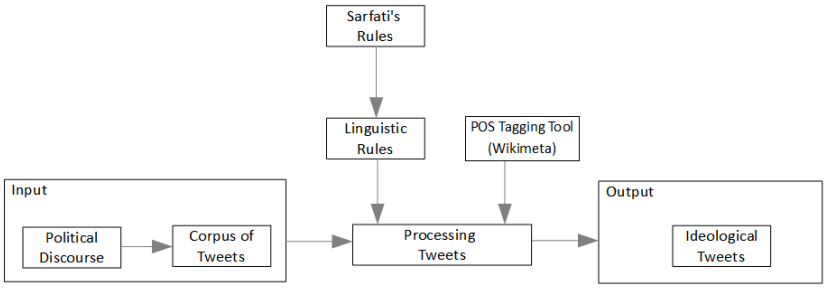


Figure 1: Ideological tweet detection system.

two rounds of the 2014 municipal elections in France. For the less active accounts we took into account even earlier tweets because we wanted to keep the density of tweets from each account and the publication rate is not the same for all; the oldest tweet was published on 2009-03-04 11:59:49).

4.3 Applying the rules

In this section we give some examples from the corpus of tweets to describe how our system processes tweets while applying the rules. It is important to recall that a tweet is identified as ideological by the system if the tweet satisfies all the 7 rules described above; note that all the 7 rule have the same weight in the system.

Tweet 1: *Je suis ravi de pouvoir compter sur tous ceux qui m'ont accompagné ce soir sur Twitter pendant #motcroises, merci à vous !*

Tweet 2: *Bruno Lemaire : "Les socialistes vivent dans le monde d'avant, c'est pourquoi nous devons inventer le monde d'après."*

Tweet 3: *Le rassemblement ce n'est pas avoir peur les uns des autres, c'est être forts ensemble.*

Tweet 4: *Ns avons perdu ms ns avons gagné un combat: faire naître l'opposition.Le dbut de l'alternance! Merci a chacune et chacun.*

Tweet 1 satisfies Rules 5, 6 and 7, but it does not satisfy Rules 1, 2, 3 and 4: Rule 1 because the tweet contains the word tonight (*ce soir* - fr), Rule 2 as it begins with the interlocution subject I (*je* - fr), Rule 3 because of the presence of the proper noun "Twitter" and Rule 4 as the tweet contains an exclamation mark.

Tweet 2 satisfies Rules 1, 2, 3, 5 and 6, but it does not satisfy Rules 4 and 7: Rule 4 because the tweets contains the modal verb must (*devons* - fr) and Rule 7 as the tweet represents a direct speech where the relator is *Bruno Lemaire* and the speech is between quotes.

Tweet 3 satisfies the 7 rules and is identified as ideological by the system: it does not contain any spatio-temporal marks or proper nouns, interlocution subjects or any connectors, exclamation or interrogation marks, modal verbs or discourse forms; moreover, the verbs' tense is the present.

Tweet 4 satisfies Rules 1, 2, 3, 5, 6 and 7, but it does not satisfy Rule 4. This tweets outlines that web-slangs and abbreviations introduce important issues in our system. Indeed Tweet 4 contains abbreviations for we (*Ns* - *nous* - fr) and for but (*ms* - *mais* - fr) wrongly annotated by Wikimeta. Thus, the system does not detect that Rules 2 and 5 are not satisfied.

However, working on a political tweets corpus ensures us that web-slangs and abbreviations are limited as politicians use proper standard French.

4.4 Results

We tested our system on 20400 tweets selected chronologically from the corpus, and 321 tweets were identified as ideological as they satisfy all 7 rules. Then, we analyzed these results from 3 points of view: (1) the 321 tweets were evaluated in order to compute the precision of our system, (2) the rest of 20079 tweets identified as non-ideological by the system were analyzed in an effort to better understand the recall of our system, and (3) we aimed to detect common linguistic patterns in the ideological tweets.

4.4.1 False positives analysis

The 321 tweets identified as ideological by the system were then manually analyzed for validation by an expert on ideology texts. The purpose

of this analysis is twofold: (1) we wanted to determine how many tweets, from the 321 identified as ideological by the system, are validated as ideological by the expert, and (2) for the tweets that are not validated as ideological by the expert, we expect to identify characteristics that would allow us to refine the results and to distinguish individual traits that can further lead us to improve our system. The result of this analysis is presented in Table 2. From the 321 tweets identified as ideological by the system, 214 tweets are validated as being ideological by the expert representing 66.66% of the 321 tweets. The rest of 33.33% is shared between tweets that are non-ideological and tweets that are partially ideological. In the following, we will detail these two categories.

For the non-ideological tweets, a detailed analyses allowed us to detect the following special cases: (1) a tweet beginning with "@" is usually a response to another tweet and, thus, it is quite brief and not ideological (e.g., *@askolovitchC il faut conduire avec moderation...*); and (2) a tweet containing "#" indicates a very specific context, thus, it cannot be interpreted independently (e.g., *#retraites : visiblement on s'oriente vers du grand n'importe quoi ...*).

The partially ideological tweets are those contextual tweets that can be interpreted out of their context and consequently become ideological. Thus, they have the specificity of allowing two interpretations: ideological and contextual. The following examples describe this type of tweets:

- the tweet *#Confsociale : l'uniformisation et la simplification des systèmes de prévention sociale et de retraite s'impose dès à présent* is contextual as it is related to a specific manifestation. Nevertheless, its content can be clearly understood outside the context.

- the tweet *@DominiqueReynie bravo pour ce travail. l'innovation est forcément une contestation de l'existant* is contextual as its author answers to another tweet, but at the same time he hopes being read by others so he adds an ideological message.

It is important to note that the expert decided to validate as ideological several tweets containing "#" or beginning with "@" as they carry strong ideological messages (e.g., *Le progrès social n'est pas l'adversaire de la performance économique #loiESS*).

4.4.2 False negatives analysis

After analyzing the set of tweets identified as ideological by the system, we also analyzed the set of tweets identified as non-ideological by the system with the aim to determine if ideological tweets have been misclassified by our system as non-ideological.

To this end, we sampled the set of tweets identified as non-ideological by the system (20079 tweets) by randomly selecting 4% of the tweets that do not satisfy only one rule (117 tweets) and 2% of the tweets falling in the other categories (329 tweets). Thus, we obtained a set of 446 tweets that was analyzed for validation by the expert. This analysis showed that 96.64% of the sampled tweets were classified correctly as non ideological, thus leaving the false negatives to represent 3.36%. One other observation is that there were no errors if a tweet does not satisfy 3 rules or more; this tweet is always correctly identified by the system as non-ideological.

Furthermore, in order to understand why these tweets were misclassified by the system, we carefully analyzed the false negatives and we made the following conclusions: (1) several misclassifications result as an error of annotation of Wikimeta; (2) several misclassifications are caused by Rule 2 as sometimes interlocution subjects (as our, *nos* - fr) are used as general referent; and (3) Rule 6 produces some misclassifications equally when the future tense dominating the tweet is prospective (e.g., *La République sera à tous les Français*). These observations will be exploited to further improve the system's performance in the future.

4.4.3 Linguistic structures identification

Analyzing the ideological tweets, the expert pointed out that they contain a style that fits into a rhetorical and strongly argumentative reference in order to give them more strength and to impose the ideology.

In this context, some structures were clearly identified:

Have to (*Il faut* - fr): e.g., *Ce qu'il faut c'est établir des priorités, choisir des filières*

Expert validation of the 321 tweets identified as ideological by the system		
Ideological tweets	Non-ideological tweets	Partially ideological tweets
214 (66.66%)	75 (23.36%)	32 (9.96%)

Table 2: Results after expert's validation of the 321 tweets identified as ideological by the system.

d'excellence, créer des emplois dans des secteurs porteurs.

There is (*Il y a* - fr): e.g., *Il y a un problème de méthode pour règler les problèmes que rencontrent nos banlieues; il faut développer des conseils de quartier élus.*

A strong syntactic structure: topicalization, such as X...is x... or which is...that is... (*X, c'est x* or *ce qui est...c'est* - fr): e.g., *Ce qui est attendu des candidats ce ne sont pas des promesses, c'est un discours de vérité sur l'effort à produire #francebleu107_1*

At the same time, the expert observed that the current hypothesis of detecting ideological tweets can be enriched with style-based criteria, which could give interesting results.

Furthermore, regarding Rule 4, it might be interesting to evaluate the tweets containing the have to verb (*devoir* - fr), as in some cases the verb have to does not necessarily indicates the involvement of the speaker, but rather a form of general truth, e.g., *Les démocrates doivent s'unir pour mettre fin à cette violence dans le débat public. #BFMTV*.

Finally, more interesting for the rest of our work would be to discriminate different types of ideologies. For example, those who do not satisfy the rule 3 may correspond to a nationalist ideology, such as *Quoi de plus naturel que l'amour de sa patrie ? Le patriotisme n'est pas un gros mot" #Souvenirfrançais*.

5 Conclusions and Future Work

In this paper, we implemented Sarfati's criteria as a set of linguistic rules for detecting ideology in textual documents. Moreover, we developed a system that implements these rules as an extension of an NLP System. Finally, we tested our system against a set of 20400 tweets of French politicians in order to experiment rules' implementation and their accuracy.

The evaluation of the rules and their implementation give us good results for the system's accuracy since 66.66% of tweets identified as ideological were indeed so and 96.64% of tweets identified as non-ideological (after sampling) were validated as non-ideological by the expert.

For the future work, we plan to take advantage of the analysis produced by the expert in order to revise or relax some of the rules that might misclassify some tweets, but also to propose a set of rules allowing us to detect the type of the ideology for those ideological tweets. Moreover, we plan to provide these rules as a standard extension to NLP systems so that they can be integrated in the everyday analysis of ideological discussions on social media.

6 Acknowledgements

This work is part of the "Digital Humanities and Data Journalisme" transdisciplinary project (funded by the Foundation of the Cergy-Pontoise University, France[8]) and of the CoMeRe project from group "Nouvelles formes de communication" of the consortium Corpus-écrits and supported by Corpus-écrits and Ortolang (Chanier et al., 2014).

References

Pierre-André Buvet. 2012. Traitement automatique du discours rapporté. In *Actes du colloque JADT 2012*.

Thierry Chanier, Céline Poudat, Benoit Sagot, Georges Antoniadis, Ciara R. Wigham, Linda Hriba, Julien Longhi, and Djamé Seddah. 2014. The comere corpus for french: structuring and annotating heterogeneous cmc genres. Submission to Journal of Language Technology and Computational Linguistics.

M Conover, B Gonçalves, J Ratkiewicz, A Flammini, and F Menczer. 2011. Predicting the political alignment of twitter users. In *Proceedings of 3rd IEEE Conference on Social Computing*.

Mats Dahllf. 2012. Automatic prediction of gender, political affiliation, and age in swedish politicians

[8] http://fondation.u-cergy.fr/

from the wording of their speeches - a comparative study of classifiability. *Literary and Linguistic Computing*, (2):139–153.

Kathleen T. Durant and Michael D. Smith. 2006. Mining sentiment classification from political web logs. In *In Proceedings of Workshop on Web Mining and Web Usage Analysis*.

Kathleen Durant and Michael Smith. 2007. Predicting the political sentiment of web log posts using supervised machine learning techniques coupled with feature selection. In *Advances in Web Mining and Web Usage Analysis*, pages 187–206. Springer Berlin / Heidelberg.

Miles Efron. 2006. Using cocitation information to estimate political orientation in web documents. *Knowledge and Information Systems*, (4):492–511.

RS Erikson and KL Tedin. 2003. *American Public Opinion*. Longman.

Sean Gerrish and David M. Blei. 2011. Predicting legislative roll calls from text. In *International Conference on Machine Learning (ICML)*, pages 489–496.

Emmanuel Giguet and Nadine Lucas. 2001. La détection automatique des citations et des locuteurs dans les textes informatifs.

Swapna Gottipati, Minghui Qiu, Liu Yang, Feida Zhu, and Jing Jiang. 2013. Predicting user's political party using ideological stances. In *Social Informatics*, pages 177–191. Springer.

Mohit Iyyer, Peter Enns, Jordan Boyd-Graber, and Philip Resnik. 2014. Political ideology detection using recursive neural networks. In *Association for Computational Linguistics*.

Kathaleen Knight. 2006. Transformations of the concept of ideology in the twentieth century. *American Political Science Review*, pages 619–626.

Julien Longhi, Claudia Marinica, Boris Borzic, and Abdul Alkhouli. 2014. Polititweets, corpus de tweets provenant de comptes politiques influents. Technical report.

Julien Longhi. 2008. *Objets discursifs et doxa : essai de sémantique discursive*. L'Harmattan.

Julien Longhi. 2013. Essai de caractérisation du tweet politique. *L'information grammaticale*, pages 125–132.

Julien Longhi. 2014. Le pigeon est-il un canard comme les autres ? esquisse dune théorie des objets discursifs. In *Res Per Nomen IV- Les théories du sens et de la référence - Hommage à Georges Kleiber*. Éditions des Presses Universitaires de Reims.

Ghassan Mourad and Jean-Pierre Desclés. 2003. Identification et extraction automatique des informations citationnelles dans un texte. *Le Discours rapporté dans tous ses états: question de frontieres?*

Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. 2010. From tweets to polls: Linking text sentiment to public opinion time series. In William W. Cohen and Samuel Gosling, editors, *Fourth International AAAI Conference on Weblogs and Social Media*. The AAAI Press.

Fabien Poulard, Thierry Waszak, Nicolas Hernandez, and Patrice Bellot. 2008. Repérage de citations, classification des styles de discours rapporté et identification des constituants citationnels en écrits journalistiques. In *Actes de la 15me Conference sur le Traitement Automatique des Langues Naturelles*.

François Rastier. 2011. *La mesure et le grain. Sémantique de corpus*. Honoré Champion, lettres numriques edition.

Yaroslav Riabinin. 2009. Computational identification of ideology in text: Study of canadian parliamentary debates. Master thesis, University of Toronto.

Georges-Elia Sarfati, 2014. *Les discours institutionnels en confrontation. Contributions a lanalyse des discours institutionnels et politiques*, chapter Lemprise du sens: Note sur les conditions theoriques et les enjeux de lanalyse du discours institutionnel, pages 13–46. LHarmattan.

Yanchuan Sim, Brice D. L. Acree, Justin H. Gross, and Noah A. Smith. 2013. Measuring ideological proportions in political speeches. In *In Proceedings of the Conference on Empirical Methods in Natural Language Processing and Natural Language Learning*.

Swapna Somasundaran and Janyce Wiebe. 2010. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, CAAGET '10, pages 116–124, Morristown, NJ, USA. Association for Computational Linguistics.

Marilyn A. Walker, Pranav Anand, Rob Abbott, Jean E. Fox Tree, Craig H. Martell, and Joseph King. 2012. That is your evidence?: Classifying stance in online political debate. *Decision Support Systems*, (4):719–729.

Daniel Xiaodan Zhou, Paul Resnick, and Qiaozhu Mei. 2011. Classifying the political leaning of news articles and users from user votes. In *Fifth International AAAI Conference on Weblogs and Social Media*. The AAAI Press.

Mapping German Tweets to Geographic Regions

Tatjana Scheffler Johannes Gontrum Matthias Wegel Steve Wendler
Department of Linguistics
University of Potsdam
firstname.lastname@uni-potsdam.de

Abstract

We present a first attempt at classifying German tweets by region using only the text of the tweets. German Twitter users are largely unwilling to share geolocation data. Here, we introduce a two-step process. First, we identify regionally salient tweets by comparing them to an "average" German tweet based on lexical features. Then, regionally salient tweets are assigned to one of 7 dialectal regions. We achieve an accuracy (on regional tweets) of up to 50% on a balanced corpus, much improved from the baseline. Finally, we show several directions in which this work can be extended and improved.

1 Introduction

Tweet collections are becoming more and more valuable as language resources due to their abundance, and the range of styles and topics they cover. Another interesting factor of Twitter data is the fact that it is much more than just text – metadata such as time stamps, user profile information and network data can be explored in NLP applications as well. Geolocation information is also sometimes present, most notably in the form of GPS coordinates of the origin of the tweet. However, while for some languages, geolocation data is commonly included in tweets, German twitterers are very reluctant to include geolocation coordinates. Of German tweets, which only make up less than 1% of all Twitter traffic, less than 2% are geo-tagged (Scheffler, 2014). In this paper, we show a data driven approach that can learn regionally salient words from seed data, and subsequently classify incoming tweets into geographic regions. Our method could be applied to other languages as well.

The aim of this study is to place German tweets geographically within a region of origin, despite the frequent lack of geolocation information. Tweets that do contain geolocation metadata (see Figure 1) are used as "gold standard" data in our work. The geolocation metadata of tweets is usually obtained from the GPS coordinates of the Twitter user (the author of the tweet) at the time of writing.

1.1 Regional expressions in tweets

Tweets that do not contain explicit geolocation metadata can still indicate where they originate from. In this first approach, we consider only the text of a tweet in order to place it geographically, and we ignore other information (for example, the authoring user and the user's given profile information). The text of a tweet can be regionally influenced in at least two ways: First, by the dialectal region of origin of the author (Twitter user). Such dialect regions could be reflected in the text by the use of regionally salient words and dialectal expressions (example (1a)). In German tweets, dialects are also often represented orthographically (e.g., by writing *ned* instead of *nicht*, 'not' example (1b)). Second, the current location of the twitterer induces the mention of location names, locally relevant person names, local events, etc

```
place (
| country =  "Germany"
| place_type =  "city"
| country_code =  "DE"
| name =  "Stuttgart"
| full_name =  "Stuttgart, Stuttgart"
| url =  "http://api.twitter.com/1/
    geo/id/e385d4d639c6a423.json"
| id =  "e385d4d639c6a423"
| bounding_box (
| | coordinates => Array (1) (
| | | ['0'] => Array (4) (
| | | | ['0'] => Array (2) (
| | | | | ['0'] =  9.038755
| | | | | ['1'] =  48.692343 )
| | | | ['1'] => Array (2) (
| | | | | ['0'] =  9.315466
| | | | | ['1'] =  48.692343 )
| | | | ['2'] => Array (2) (
| | | | | ['0'] =  9.315466
| | | | | ['1'] =  48.866225 )
| | | | ['3'] => Array (2) (
| | | | | ['0'] =  9.038755
| | | | | ['1'] =  48.866225 ) ) )
| | type =  "Polygon" )
| attributes ( )
)
```

Figure 1: Geolocation metadata of a tweet (JSON).

(example (2)). Both kinds of regional influences on tweet texts can of course pertain at the same time and possibly independently of each other, as when a person from Bavaria (region of origin) visits Berlin (current location). In this case, a mix of Bavarian terms and Berlin-specific names may occur.

(1) a. *Jep, der Lütte ist inzwischen 4,5 Jahre alt. . . .*
Yup, the little-one [regional Northern term] is now 4.5 years old. . . .
b. *Weiß ned, was ich lustiger finde. . .*
Don't know what's funnier to me. . .

(2) *Falls ihr jemanden mit einer Zwergenmütze durch Berlin laufen seht- winkt mir doch!*
If you see anyone walking through Berlin with a gnome hat, wave at me!

Although both kinds of regional influences are partially independent of each other, in this first attempt we have not tried to tease them apart systematically. Instead, we take geo-tagged tweets as accurately reflecting their origin and try to recover this geographic information in untagged tweets. Our basic assumption is that regionally diverging tweets (where regional origin and current location don't match) should be relatively rare compared to converging tweets, so that the basic signal does not get obscured for machine learning. In addition, our probabilistic model of regional salience (introduced below) allows for tweets and lexical items to be associated with several regions at the same time. With enough training data (and ignoring sparse data problems for the moment), this would allow for a tweet to be identified as associated with Bavaria and Berlin in equal measure.

1.2 German dialect regions

In this work, we defined dialect regions by hand based on existing classifications. For this purpose, we split the German-speaking European area into seven non-overlapping regions, along dialectal and structural boundaries (see Figure 2). We determined the regions based on the data in the *Atlas zur deutschen Alltagssprache* (de Liege and Salzburg, 2013). We also had to take some Twitter-specific properties into account. For example, the data of the *Atlas* also showed a small region around Saarland and Luxemburg to have characteristic idiosyncrasies, but we did not split it off because there would be too few tweets from such a small region.

1.3 Outline of this paper

In the following section, we give a brief overview of previous work with regard to processing German Twitter data and geolocation data encoded in tweets. Section 3 presents the data used in this work. Subsequently, we discuss our approach to finding the geographical origin of tweets and present our results. In the final section, we discuss the approach used and present several possible directions for further research.

2 Related Work

2.1 German Twitter

There is very little previous work on German Twitter data. Social media NLP research has largely concentrated on English, because English data are much more abundant (about 40–50% of

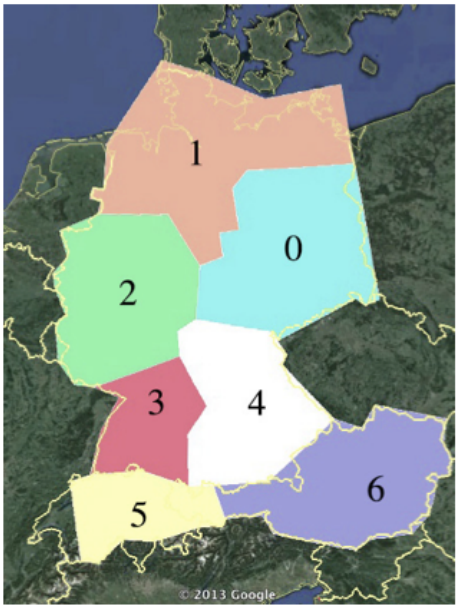


Figure 2: Map of the regions and the index of their feature used in the vectors represented as polygons.

all tweets) and thus easier to obtain. (Scheffler, 2014) introduces a large-scale corpus of German tweets, part of which is used in this work. Scheffler shows that in her corpus, which is an almost complete collection of all German-language tweets sent in April, 2013, less than 2% of these tweets contain public geolocation metadata.

There has been some work on adapting common NLP applications to German Twitter data, such as POS tagging (Rehbein et al., 2013b) and normalization (Sidarenka et al., 2013). And though certain linguistic phenomena have been studied using German Twitter data, including the specific style present on Twitter (Rehbein et al., 2013a), to our knowledge, no previous work has analysed the geographic origin or distribution of German tweets.

2.2 Tweets and geolocation

For other languages, the relationship between tweets and their location of origin has been looked at in several different ways. For example, (Arakawa et al., 2011) propose a three-tier search algorithm to find location dependent words. Their goal is to find place names and other terms (e.g., store names) to aid a predictive Japanese text-entry system. (Eisenstein et al., 2012) present a sociolinguistic study and model that shows how neologisms spread between US cities based on tweets. They used only data which included public geo-tags, while (Arakawa et al., 2011) devised a method to find geographically anchored Twitter data, even when those geo-tags are set to "private" by the users (they still show up in geographic Twitter searches). Recent work by (Grieve, 2014) on the regional distribution of variants in English also makes use of tweets with geolocation metadata.

Previous work on localizing tweets has for example built on language models (Kinsella et al., 2011), and has often tried to classify the location of users instead of a single tweet (Cheng et al., 2010; Hecht et al., 2011). In a different approach, (Leetaru et al., 2013) applied an algorithm developed for geocoding Wikipedia articles (Leetaru, 2012) to tweets. Since this approach is based on finding explicit location names in the text, it cannot be used to find the geographic origin of the vast majority of tweets.

3 Data

Our study is based on a corpus of German tweets collected in April 2013. It was collected by filtering the Twitter stream using a list of 397 common German words as key words (any tweet containing any word on the list is returned). The filtered stream was further narrowed down using the language identification module LangId (Lui and Baldwin, 2012), which yields very good results for our German data. The remaining data covers upwards of 90% of all German-language tweets sent during that period. We collected on average about 800,000 German tweets per day, for a total of 24,179,872 (see (Scheffler, 2014) for more detail on the corpus and the collection method). Out of these, only 254,874 tweets contained geolocation attributes. We eliminated tweets authored by two spam bots, all retweets, as well as automatically created tweets with the hashtags "#nowplaying", "#np", and "#4sq". After holding out 150 tweets from each region as a test set, the remaining 174,011 tweets formed our training corpus (geo-174k).

Since the regions were not represented equally in the training data (the smallest region, Austria, had only 8637 tweets, excluding the test set),

we built several balanced sub-corpora to measure the influence of the size of the training corpus: balanced-60k (the maximal balanced corpus with 60,459 tweets), balanced-21k with 3000 tweets from each region, and balanced-39k, all 39,459 tweets in the former sub-corpus but not the latter.

We performed almost no pre-processing on the data beyond the filtering described above. The tweets were tokenized using Christopher Potts' Twitter tokenizer[1], which recognizes such social media-specific entities as URLs, emoticons, etc. The resulting tokens were converted to lower case, yielding the final list of tokens for each tweet.

4 Geo-Mapping German Tweets

Our basic method is to represent each word in a corpus of tweets as a region vector representing the probability of that word originating from that region. Following the two kinds of regional influences on language mentioned above, we devised two approaches to train the initial region vectors from our training data: an approach based on dialectal expressions found in the Atlas zur deutschen Alltagssprache, and one trained directly from tweets that are tagged with geolocation information.

4.1 Regional words approach

The first attempt uses a seed word list of hand-selected regional expressions. As a source for the regional expressions we used the Atlas zur deutschen Alltagssprache (de Liege and Salzburg, 2013), which contains maps aggregating survey data on dialectal variants.

We included terms from the Atlas based on the following factors. Variants not reflected in the written form (such as vowel qualities) were excluded, as were multi-word expressions (e.g., *viertel vor*, a variant for the temporal expression 'quarter to'). We also excluded terms that showed too much overlap (did not adhere to clear dialect boundaries) or covered almost the entire language area (e.g., *Backofen*, 'oven'). A word was only included in our seed list of regional terms if it appeared in a maximum of four out of our seven regions. Furthermore, homonyms and polysemes were inappropriate for our purposes, so for example most of the regional words for 'attic', including *Boden*, *Speicher* and *Bühne*, were ruled out. We also went without very short expressions like *wa* (Berlin dialect for the question tag 'right?') because of the high chance of coincidence with abbreviations and cropped words.

In total, we selected a list of 209 regionally dependent terms from the Atlas, and split the probability mass uniformly between the regions in which the term is attested in order to yield seed vectors. E.g., the region vector for *Porree* ('leek') is (.33 .33 .33 0 0 0 0), since this word is only used in East, North, and West Germany (in the South, the variant *Lauch* is used). The disadvantage of this approach is the sparseness of the data, especially with regard to the kinds of terms not found in the Atlas (which contains mostly food related and outdated terms).

4.2 Training from geolocated tweets

In the second approach, we trained the seed vectors directly from tweets that have been tagged with GPS geolocation metadata by their authors. We used the following algorithm (Algorithm 1) to assess the probabilities of a certain term originating from a certain region by directly observing geo-tagged training data. For each tweet, we determined its originating region using the point-in-polygon algorithm from (Lawhead, 2011) and initialize the tweet vector as 1 for the originating region, and 0 for all others. For each term in the tweet, excluding stop words, we then added this tweet vector to the word vector for the term. After all tweets in the training corpus have been processed, these word vectors (essentially, counts of how often a word originated from each region) were then normalized to yield probabilities.

Following the initialization of the word vectors by one of the above methods, we included a bootstrapping step during which the vectors could be adjusted using additional data without geolocation information. In a nutshell, first a tweet vector is calculated for each tweet in the bootstrapping corpus based on the existing generation's word vectors (classification), and then a new generation of word vectors is calculated for the corpus based on the tweet vectors for all the tweets that a particular word occurs in (bootstrapping step).

[1] http://sentiment.christopherpotts.net/code-data/happyfuntokenizing.py

Data:
tweets: Corpus of geo-annotated documents
stopwords: List of stopwords
Result:
WV: normalized word vectors, representing the probability distribution for each word

$WV \leftarrow \emptyset$;
foreach *tweet in tweets* **do**
  $region \leftarrow \texttt{Classify}(tweet)$;
  $\vec{tweet} \leftarrow \texttt{CreateVector}(region)$;
  forall *token in tweet* **do**
    if *token* $\notin$ *stopwords* **then**
      $WV(token) \leftarrow WV(token) + \vec{tweet}$;
    end
  end
end
foreach $\vec{word}$ *in WV* **do**
  $\vec{word} \leftarrow \texttt{normalize}(\vec{word})$;
end
return *WV;*

Algorithm 1: Obtaining regional probabilities for words.

Finally, after training and bootstrapping, the word vectors can be used to classify tweets into regions. For classification, we used the cosine similarity between the tweet vector and the average tweet vector over the entire bootstrapping corpus. A tweet would be assigned to the dimension (region) in which the difference vector between the current tweet vector and the average tweet vector is maximal. Note however, that a huge majority of German tweets are written in standard German without any signs of regional influence whatsoever, or are very short. In order to alleviate this problem, we used a variable threshold of "non-regional tweets", below which we did not attempt to classify a tweet. This threshold (called "guess" in Algorithm 2) was set experimentally as the minimum difference (maximum cosine similarity) between a tweet vector and the average tweet vector, reasoning that a tweet that is very similar to the average of all tweets doesn't show any clear regional trends. Algorithm 2 computes the "average tweet" vector to compare each tweet with during classification, as well as the cosine similarity threshold beyond which a tweet is recognized as sufficiently "different" from the average. This threshold is computed based on a preset percentage of assumed regional tweets. We

Data: *tweets*: Set of geo-annotated documents
guess: guessed percentage of regional tweets
WV: Set of word vectors
Result: *threshold*: cosine similarity threshold

$tweetvectors \leftarrow \emptyset$;
foreach *tweet in tweets* **do**
  $\vec{tweet} \leftarrow (0,0,0,0,0,0,0,0)$;
  forall *token in tweet* **do**
    if $token \in WV$ **then**
      $\vec{tweet} \leftarrow \vec{tweet} + WV(token)$;
    end
    $tweetvectors \leftarrow tweetvectors \cup \{\vec{tweet}\}$;
  end
end
$\vec{average} \leftarrow (0,0,0,0,0,0,0,0)$;
foreach $\vec{tweet} in tweetvectors$ **do**
  $\vec{average} \leftarrow \vec{average} + \vec{tweet}$;
end
$\vec{average} \leftarrow \frac{\vec{average}}{l(tweetvectors)}$;
$vectorlist \leftarrow \emptyset$;
foreach $\vec{tweet}$ *in* $tweetvectors$ **do**
  $similarity \leftarrow sim(\vec{tweet}, \vec{average})$;
  $vectorlist \leftarrow append(similarity)$;
end
$vectorlist.sort()$;
$threshold = vectorlist[int(guess \times l(vectorlist))]$;
return *threshold*;

Algorithm 2: Cosine similarity algorithm.

discuss below how this threshold is set.

The final parameter influencing the results is the length of the stop word list. We compiled a custom stop word list by excluding the most frequent N words in the training corpus. The best value for N was determined experimentally.

5 Results

Here, we first report the results of the approach using geo-tagged data for estimating the initial word vectors. A naïve random baseline for tweet classification on the balanced test set should yield an accuracy of 1/7 = 0.14 for seven regions.

First, we evaluated the best data set combinations for the training and bootstrapping stage; all numbers are accuracy scores on the held-out test set of 1050 tweets (150 from each region). For subsequent experiments, we used the best data sets determined above: For training, the balanced-39k corpus, and for any bootstrapping steps, the

entire (unbalanced) geo-tagged corpus of 174k tweets.

Next, we assessed the effect of the number of stop words excluded. Figure 3 shows that performance decreases again after 200 words, maybe because some regional words are very common.

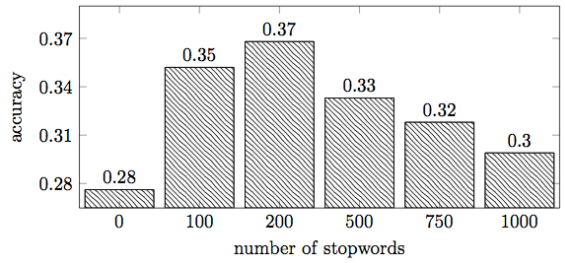


Figure 3: Accuracy based on size of stop word lists.

To determine the optimal cosine similarity threshold ("guess") to distinguish "standard German" from regional tweets, we varied the number of regional tweets we attempted to classify in steps of 10%. Clearly, the accuracy rises the fewer tweets are deemed "regionally salient". The optimal result on the test set is reached with only 20% of tweets deemed sufficiently different from the average to be classified. The overall accuracy on this setting reaches 0.506 (see Figure 4).

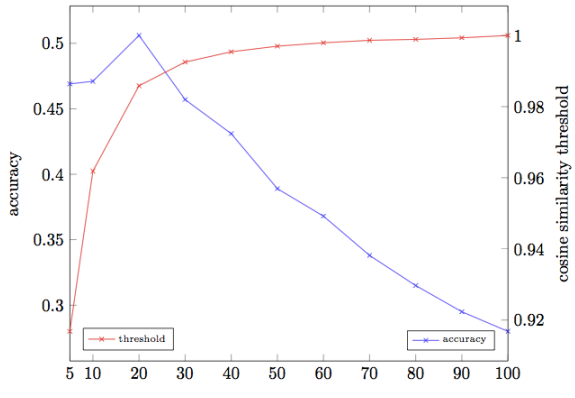


Figure 4: Relation between percentage of regional tweets and accuracy.

Finally, we estimated the effect of the number of bootstrapping loops included in the calculation. Any number of bootstrapping steps actually decreases the overall accuracy. We suspect that this happens because during bootstrapping, all vectors are assimilated more and more to the average vector.

The best result of our classification algorithm is obtained with the balanced-39k training corpus and the geo-174k corpus used in order to compute the overall average tweet vector (the bootstrapping step is skipped), with 200 stop words excluded and 20% of tweets deemed regionally salient (this corresponds to a maximum cosine similarity value of 0.94). With these settings, we achieve an accuracy of 0.53 on the test set.

Using the regional words approach, the results were much worse, reaching only up to an accuracy of 0.3 in the best case. We kept the percentage of regional tweets (20%) and the stop word list (200 words) constant.

6 Discussion

In this paper, we have shown a data-driven method to regional classification of German tweets. Our approach is trained on a medium-sized corpus of geographically tagged German tweets by deriving regional probabilities for each word in the corpus. Though most tweets are standard German and cannot be assigned to one particular region, we automatically identify the 20% most significantly regionally influenced tweets. Our classification accuracy on these 20% is 0.53 with optimal settings, a significant improvement over the 0.14 random baseline.

Our second approach based on a seed set of regionally salient words yields a much lower accuracy of less than 0.3 due to sparse data problems. An obvious idea for future work is the combination of the two methods, since they capture different intuitions: the geolocation metadata used in the geolocated tweets approach is based on the current location of the twitterer (usually, GPS location obtained from a mobile phone). In contrast, the regional and dialectal expressions covered in the Atlas zur deutschen Alltagssprache more likely reflect the regional origin of the twitterer (no matter her/his current location). It could also be worthwhile to amend the regional word seed list, which is currently very small (only 209 terms). Then, it could be combined with additional geo-tagged Twitter data in a bootstrapping step as outlined above.

In addition, the current scoring scheme is very rigid and does not reflect the fact that some regions are more similar to each other than others,

	Assigned region						
True region	*0*	*1*	*2*	*3*	*4*	*5*	*6*
0 = East	.18	.41	.12	.06	.00	.06	.18
1 = North	.12	.65	.00	.12	.00	.06	.06
2 = West	.09	.23	.45	.14	.05	.05	.00
3 = Southwest	.04	.22	.13	.52	.09	.00	.00
4 = Bavaria	.05	.29	.05	.00	.57	.00	.05
5 = Switzerland	.02	.16	.04	.08	.00	.68	.02
6 = Austria	.05	.27	.00	.05	.05	.18	.41

Table 1: Confusion matrix for final run.

as is also visible from the confusion matrix in Table 1. The table also indicates that most misclassifications are assigned wrongly to region 1 (North), indicating a problem with the definition of that region or with the corpus training data we have for it.

Another obvious extension to the work reported here, as suggested by one of the reviewers, is a qualitative evaluation of regional and non-regional German tweets with respect to linguistic and lexical features. This may lead to an improved regional seed word list, possibly a new region assignment, and new insights for the localization of tweets.

Acknowledgments

We are very grateful to the four reviewers' detailed comments and questions. This work has been supported by the collaborative project "Analysis of Discourse in Social Media" (project number 01UG1232A), funded by the German Federal Ministry of Education and Research.

References

Yutaka Arakawa, Shigeaki Tagashira, and Akira Fukuda. 2011. Spatial statistics with three-tier breadth first search for analyzing social geocontents. In A. König, A. Dengel, K. Hinkelmann, K. Kise, and R. J. Howlett, editors, *Proceedings of the 15th international conference on Knowledge-based and intelligent information and engineering systems (KES'11)*, volume Part IV, pages 252–260. Springer.

Zhiyuan Cheng, James Caverlee, and Kyumin Lee. 2010. You are where you tweet: A content-based approach to geo-locating Twitter users. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pages 759–768, New York, NY, USA. ACM.

Universite de Liege and Universität Salzburg. 2013. Atlas zur deutschen Alltagssprache. online resource. http://www.atlas-alltagssprache.de/.

Jacob Eisenstein, Brendan O'Connor, Noah A. Smith, and Eric P. Xing. 2012. Mapping the geographical diffusion of new words. *CoRR*, abs/1210.5268.

Jack Grieve. 2014. A comparison of statistical methods for the aggregation of regional linguistic variation. In Benedikt Szmrecsanyi and Bernhard Wälchli, editors, *Aggregating dialectology, typology, and register analysis: Linguistic variation in text and speec*. Walter de Gruyter, Berlin/New York.

Brent Hecht, Lichan Hong, Bongwon Suh, and Ed H. Chi. 2011. Tweets from Justin Bieber's heart: The dynamics of the location field in user profiles. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 237–246, New York, NY, USA. ACM.

Sheila Kinsella, Vanessa Murdock, and Neil O'Hare. 2011. "I'm eating a sandwich in Glasgow": Modeling locations with tweets. In *Proceedings of the 3rd International Workshop on Search and Mining User-generated Contents*, SMUC '11, pages 61–68, New York, NY, USA. ACM.

Joel Lawhead. 2011. Point in polygon 2: Walking the line. http: //geospatialpython.com/2011/08/point-in-polygon-2-on-line.html.

Kalev Leetaru, Shaowen Wang, Guofeng Cao, Anand Padmanabhan, and Eric Shook. 2013. Mapping the global Twitter heartbeat: The geography of Twitter. *First Monday*, 18(5).

Kalev Leetaru. 2012. Fulltext geocoding versus spatial metadata for large text archives: Towards a geographically enriched Wikipedia. *D-Lib Magazine*, 18(9):5.

Marco Lui and Timothy Baldwin. 2012. langid.py: An off-the-shelf language identification tool. In *ACL (System Demonstrations)*, pages 25–30. The Association for Computer Linguistics.

Ines Rehbein, Sören Schalowski, Nadja Reinhold, and Emiel Visser. 2013a. Uhm... uh.. filled pauses in computer-mediated communication. Talk presented at the Workshop on "Modelling Non-Standardized Writing" at the 35th Annual Conference of the German Linguistic Society (DGfS).

Ines Rehbein, Emiel Visser, and Nadine Lestmann. 2013b. Discussing best practices for the annotation of Twitter microtext. In *Proceedings of the Third Workshop on Annotation of Corpora for Research in the Humanities (ACRH-3)*, Sofia, Bulgaria.

Tatjana Scheffler. 2014. A German Twitter snapshot. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Uladzimir Sidarenka, Tatjana Scheffler, and Manfred Stede. 2013. Rule-based normalization of German Twitter messages. In *Proc. of the GSCL Workshop Verarbeitung und Annotation von Sprachdaten aus Genres internetbasierter Kommunikation*.

Detecting Ironic Speech Acts in Multilevel Annotated German Web Comments

Bianka Trevisan[1], **Melanie Neunerdt**[2], **Tim Hemig**[1], **Eva-Maria Jakobs**[1], **Rudolf Mathar**[2]
Textlinguistics and Technical Communication[1],
Institute for Theoretical Information Technology[2],
RWTH Aachen University, Germany

Abstract

Ironic speech act detection is indispensable for automatic opinion mining. This paper presents a pattern-based approach for the detection of ironic speech acts in German Web comments. The approach is based on a multilevel annotation model. Based on a gold standard corpus with labeled ironic sentences, multilevel patterns are determined according to statistical and linguistic analysis. The extracted patterns serve to detect ironic speech acts in a Web comment test corpus. Automatic detection and inter-annotator results achieved by human annotators show that the detection of ironic sentences is a challenging task. However, we show that it is possible to automatically detect ironic sentences with relatively high precision up to 63%. [1]

1 Introduction

Automatic detection of irony in text is a challenging task. However, typical characteristics, e.g., emoticons, inherent in Web comments, are strong indicators for ironic speech acts. This forms a new basis for the detection of irony. In this paper, we present a pattern-based approach for the detection of ironic speech acts in German Web comments. Challenges in the identification of ironic speech acts concern the fact that the identification of irony without the context is almost impossible (Sandig, 2006). Hence, sophisticated techniques are required that allow for irony detection (Mihalcea and Strapparava, 2006). For Web comments, however, typical characteristics or indicators of ironic speech acts are identified such as winking emoticons (Neunerdt et al., 2012), quotation marks, positive interjections (Carvalho et al., 2009) or opinionated words (Klenner, 2009). In contrast to standardized texts, we believe that in Web comments such characteristics allow for better detection of ironic speech acts. Nevertheless, the question is, can ironic speech acts reliably and automatically be detected based on these indicators in Web comments and what challenges arise?

Contrary to the common conceptualization, we assume that ironic speech acts are not only characterized by features at the text surface but rather by a whole set of linguistic means whose specific combination (*pattern*) indicates a specific speech act such as *IRONIZE*. In order to identify and define these patterns, we suggest a fine-grained multilevel annotation model where different linguistic means are considered. The annotation on different levels allows for level-vise and level-combined pattern analysis. The proposed approach works as follows.

First, based on a gold standard Web comment corpus typical ironic multilevel patterns (training patterns) are determined according to statistical and linguistic analysis for the detection of ironic speech acts. The gold standard corpus is manually annotated on all annotation levels. Second, the revealed training patterns serve to detect ironic speech acts in a huge Web comment test corpus. The test corpus is tokenized and Part-of-Speech

(POS) tagged automatically by the *WebTagger* proposed in (Neunerdt et al., 2013a). Based on the tokens and POS tags, the Web comments are labeled on multiple annotation levels by the *AutoAnnotator* (Trevisan et al., 2014). Detection results achieved with the training patterns are manually annotated by different annotators and evaluated.

The paper is structured as follows. Section 2 summarizes related work on irony conceptualization and detection. In Section 3, we introduce the multilevel annotation scheme and the pattern detection method. Section 4 reports the different corpora and experimental results. They are discussed in Section 5. In Section 6 we conclude our work and outline future work.

2 Related Work

In linguistics, there is a huge research regarding speech act theory. In our work, we follow the approach of (Sandig, 1979) who focuses on specific speech acts, namely evaluative speech acts such as ironic speech acts (*linguistic evaluation theory*). (Sandig, 1979), and in the following (Ripfel, 1987), conceptualizes the process of evaluation, respectively, an evaluative speech act as an act in which a subject evaluates an object with a specific purpose using evaluative expressions or linguistic means such as idiomatic expressions (e.g. *Too many cooks spoil the broth*), attributes (e.g. *right* vs. *wrong*) or evaluative lexis (e.g. *brick*) (Trevisan and Jakobs, 2010; Trevisan, 2014). The linguistic means can be used for different evaluative purposes, such as stylistic and pragmatic means for the purpose of *addressee-oriented evaluation*. In this kind of evaluation, the speaker formulates and modifies speech acts according to the evaluative intention of the communication situation and the addressee. The modification of the speech act is done by changing the style or manner of formulation. Possible speech acts are, for example, *IRONIZE*, *STRENGTHEN*, or *WEAKEN*.

Thereby, irony is an extremely complex or form-rich speech act, exemplified by the fact that multiple linguistic means are used for different phenomena, such as argument something ad absurdum, reverse something, or explicate logical relationships too clearly (Bohnes, 1997). In addition, challenges in the detection of ironic speech acts relate, particularly, to the strong interpretive ductus and context-dependency. Hence, regarding the focus of this paper, the automated detection of ironic speech acts in Web comments, the challenging task is to deal with different forms of irony and to find out which indicators are most useful for irony detection.

In computational linguistics, there is initial work done regarding the automated detection of irony in text. Approaches in this context mainly focus on the identification of emotions or humor. (Carvalho et al., 2009) identified surface clues of positive ironic sentences in comments applying a rule-based approach. In this approach, patterns are defined whose occurrence shows evidence of certain surface clues, e.g., the pattern *(ADJpos|Npos)* as indicator for irony by quotation marks. The authors found out that irony-indicating surface characteristics in sentences with a positive predicate are besides quotation marks, onomatopoetic expressions, heavy punctuation marks, and positive interjections. (Mihalcea and Strapparava, 2006) used automatic classification techniques to identify humour in one-liners, i.e., short sentence characterized by simple syntax, use of rhetoric means (e.g. alliteration), and creative language constructions. The results show that it is possible to distinguish humorous and non-humorous sentences, but the technique failed regarding the automatic and reliable identification of irony. Therefore, more sophisticated techniques are needed.

Beyond the reported approaches, there are several more in computational linguistics that provide hints on indicators of ironic speech acts in different text types. For instance, winking emoticons (*;)* and *;-)*) are irony indicators especially in chat communication (Beißwenger et al., 2012) and Web comments (Neunerdt et al., 2012). (Klenner, 2009) points out that in prose texts a positive attributive adjective and a negative noun ($ADJA^{+}$ NN^{-}) indicate an ironic speech act.

However, all described approaches do not provide a full-automated solution for the detection of ironic speech acts.

3 Methodological Approach

To detect ironic speech acts in Web comments, different indicators of multiple linguistic levels

are considered and subsumend into patterns. The multilevel annotation is described in Section 3.1, the methodology for pattern-based detection of ironic speech acts in Section 3.2.

3.1 Multilevel Annotation

In order to define patterns for detection, a linguistic multilevel annotation model proposed by (Trevisan, 2014) is applied. In the model, Netspeak-specific pecularities are considered and modeled such as non-standard parts of speech (e.g. Leetspeak), interaction signs (e.g. emoticons), different speech acts (e.g. *IRONIZE*) or syntactic peculiarities of Web language such as missing punctuation marks (Trevisan, 2014). Totally, the model contains seven linguistic annotation levels (graphematic, morphological, syntactic, semantic, pragmatic and polarity level, level of rhetorical means) and its sub-levels. At each level, different linguistic means are annotated, for instance, at the *pragmatic or target level* 30 different speech acts. The annotation model is based on the assumption that the annotated linguistic means and levels provide evidence or clues for the detection of evaluative speech acts in Web comments.

In this approach, we particularly consider *ironic speech acts* as target class. For the detection of ironic speech acts, three annotation levels out of seven are selected: POS level, graphematic level, and token polarity level. These levels are chosen due to the fact that a tool exists to annotate such levels automatically (*AutoAnnotator*) (Trevisan et al., 2014). We assume that indicators of these automatically annotated levels are mutually dependent in their appearance and, thus, in combination turn into patterns that can be more or less reliably used for the automatic detection of ironic speech acts. As speech act boundaries, we consider the beginning and the end of a sentence, determined by the corresponding POS tag on POS level.

Hereafter, the annotation levels used for pattern creation are described briefly in chronological order. Note that the terms label and tag are used synonymously.

- *Level 1 - POS level* (l_1): At the POS level, to each token a morphosyntactic category is assigned providing information about part of speech and syntactic function. POS tags are assigned according to the Stuttgart-Tuebingen Tagset (STTS), and lemma information according to a special lexicon (Schmid, 1995); (Schiller et al., 1999). In total, the tagset consists of 54 tags. Since the tagset was developed on standard texts such as newspaper articles, tag correspondences had to be defined for Netspeak-specific expressions such as emoticons (EMO = $.) (Trevisan et al., 2012); (Neunerdt et al., 2013b).

- *Level 2- Graphematic level* (l_2): At the graphematic level, expressions at the text surface as well as grapho-stilistic features that show special notational styles are annotated following (Gimpel et al., 2011). In total, eight labels are distinguished: adressing terms (e.g. *@[John], 2[heise]*; label: ADD), words with capital letters within (e.g. *CrazyChicks*; label: BMAJ), emoticons (e.g. *;-)* ; label: EMO), iterations (e.g.*yeeeeeees* ; label: ITER), leetspeak (e.g. *W1k1pedia* ; label: LEET), words in capital letters (e.g. *GREAT*; label: MAJ), markings (e.g. **[quiet]**; label: MARK) and mathematical symbols (e.g. +; label: MAT).

- *Level 3 - Token polarity level* (l_3): At the level of token polarity, the polarities of individual tokens are annotated, i.e., the polarity of words or interactive signs. There are five categories distinguished: negative token (e.g. *harmful*; label: $-$), positive token (e.g. *suitable*; label: $+$), deminisher (e.g. *less*; label: $\%$), intensifier (e.g. *much*; label: $\wedge$) and reverser (e.g. *not*; label: $\sim$).

3.2 Pattern-based Detection

The goal of our work is to detect ironic speech acts in Web comments. The overall approach is simple, based on statistical and linguistic criteria. Training patterns are defined based on a gold standard corpus, which are later used to detect sentences representing ironic speech acts (*ironic sentences*) in a Web comment corpus. In the following, we mathematically describe the two steps of our approach: First, we describe the identification

of frequent patterns over multiple annotation levels in the gold standard corpus and, second, the search process of the defined patterns for the detection of ironic speech acts in the test corpus. Therefore, we consider the gold standard corpus consisting of K sentences with labeled ironic sentences. Note that the sentence boundaries are determined by the corresponding POS tag information. Each sentence $k \in K$ contains a sequence of N_k tokens:

$$(w_1, \ldots, w_{N_k}) \in \mathcal{W}^{N_k}$$

where $\mathcal{W}$ contains all possible tokens. For each annotation level $l = 1, \ldots, L$, the corresponding labels

$$\left(t_1^l, \ldots, t_{N_k}^l\right) \in \left(\mathcal{T}_l \cup \{\epsilon\}\right)^{N_k}$$

are assigned, where $\mathcal{T}_l$ represents the set of L_l labels for a particular annotation level l:

$$\mathcal{T}_l = \left\{c_1^l, \ldots, c_{L_l}^l\right\}.$$

In our approach, we consider $L = 3$ levels, e.g., the token polarity level with $\mathcal{T}_3 = \{+, -, \%, \wedge, \sim\}$ as described in Section 3.1. Note that on some levels it is not mandatory to annotate each token. Hence, tokens which are not annotated are labeled with ϵ. The gold standard corpus labels are assigned manually by human annotators. The test corpus is labeled by means of *AutoAnnotator*, which is described in Section 3.1.

In order to determine frequent patterns in the gold standard, we first determine the label combinations of a sentence. First, for each level a feature vector

$$\mathbf{m}^l = \left(m_1^l, \ldots m_{L_l}^l\right) \tag{1}$$

with

$$m_p^l = \begin{cases} 1 & \exists\, n : t_n^l = c_p^l \\ 0 & elsewise \end{cases}$$

is calculated. The single components m_p^l indicate the presence (1) or absence (0) of a particular label c_p^l. These feature vectors are determined for all sentences $k \in \mathcal{K}$ as $\mathbf{m}_k^l$. Exemplarily, for the sentence k: *"Schon mal zu optimistisch an ein Projekt ran gegangen ;o)?" ("Have you ever tackled a project too optimistic ;o?")*, the feature vector, e.g., for level 3, results in $\mathbf{m}_k^3 = (1, 0, 0, 1, 0)$.

In order to detect statistical peculiarities, we determine the frequency of all occurring label combinations for single level, tuples and triples of levels, i.e., for n levels $l_1, \ldots, l_n \in \{1, \ldots, L\}$ and jointly occurring feature vectors $\mathbf{m}^{l_1}, \ldots, \mathbf{m}^{l_n}$ we calculate

$$N(M^{\mathbf{P}}) = \left|\left\{k \in \mathcal{K} \mid \mathbf{m}_k^{l_i} = \mathbf{m}^{l_i}, \forall i = 1, \ldots, n\right\}\right|$$

with

$$\mathbf{P} = \{l_1, \ldots, l_n\}$$

and

$$M^{\mathbf{P}} = \left(\mathbf{m}^{l_1}, \ldots, \mathbf{m}^{l_n}\right).$$

Tuples and triples are in the following sorted according to their frequencies. Example tuples and triples are given in the forth column of Table 1. According to the top frequencies and considering the pattern frequency in ironic speech acts (*IRONIZE*) only $N_I(M^{\mathbf{P}})$ compared to their frequency in other speech acts a set of tuples and triples is selected. The selected patterns fullfill $N_I(M^{\mathbf{P}})/N(M^{\mathbf{P}}) \geq 0.8$ and serve for further linguistic analysis. Based on the qualitative results, some tuples and triples are slightly modified or added due to the results, see Section 4.

The extracted tuples and triples serve to detect ironic sentences in a test corpus. The test on an arbitrary sentence works as follows. First, we calculate its feature vectors M_t according to (1). A sentence t is declared ironic if one of the defined training patterns $M^{\mathbf{P}}$ fulfills the equation

$$\text{IRONIC}(M^{\mathbf{P}}, M_t) = \prod_{l \in \mathbf{P}} I(\mathbf{m}^l, \mathbf{m}_t^l) = 1$$

with

$$I(\mathbf{m}^l, \mathbf{m}_t^l) = \prod_{p=1,\ldots,L_l} IM(m_p^l, m_{t,p}^l),$$

i.e., on each level $l \in \mathbf{P}$ at least the labels seen in the training pattern have to be present. Hence, we define

$$\text{IM}(m_p^l, m_{t,p}^l) = \begin{cases} 1 & m_p^l \leq m_{t,p}^l \\ 0 & elsewise. \end{cases}$$

We use the minimum criteria fit instead of an exact match in order to relax the restrictions. For example, on the POS annotation level an exact pattern match would lead to very strong restrictions.

4 Experimental Results

The aim of our paper is the identification of indicators and patterns that allow reliable automatic detection of ironic speech acts in Web comments. To this end, we first search for indicators of ironic speech acts in a multilevel annotated gold standard corpus (Section 4.1). In a second step, the extracted patterns are used to detect ironic speech acts in the Web comment test corpus and extract the corresponding sentences (Section 4.2).

4.1 Corpora

As an exemplary corpus, a topic-specific Web comment corpus is collected from *Heise.de*, which is a popular German newsticker site treating different technological topics. Web comments from 2008 and 2009 are collected. In total, the *Heise* corpus contains approximately 15 Million tokens.

For training purposes, a small corpus *HeiseTrain* containing Web comments with approximately 36,000 tokens is separated according to different criteria. The remaining Web comments serve as test corpus (*HeiseTest*) to evaluate the sentence extraction according to patterns for ironic speech acts (see Section 3.2). *HeiseTrain* serves as gold standard, which is manually annotated on multiple levels according to Section 3.1, among others the target level with labeled ironic sentences. For manual multilevel annotation, the tool *EXMARaLDA* is used, which is formally applied for conversational research, e.g., the analysis of audio transcripts. The annotation is performed by five annotators (Trevisan, 2014). Annotator 1-4 annotate at all levels the entire corpus. Annotator 5 annotates only those text segments, where no majority decision could be determined between annotator 1-4. Finally, the gold standard is derived from the annotation of annotator 1-5.

Figure 1 shows the corpus statistics for the target level on which evaluative speech acts are annotated. Additionally, l_1 (POS level), l_2 (graphematic level) and l_3 (token polarity level) statistics are given for the 220 ironic speech acts (*IRONIZE*) exclusively. As evident from the statistics for target level, the top 5 ranked speech acts reach more than half of all identified speech acts. Therein, the speech act *IRONIZE* (n=220)

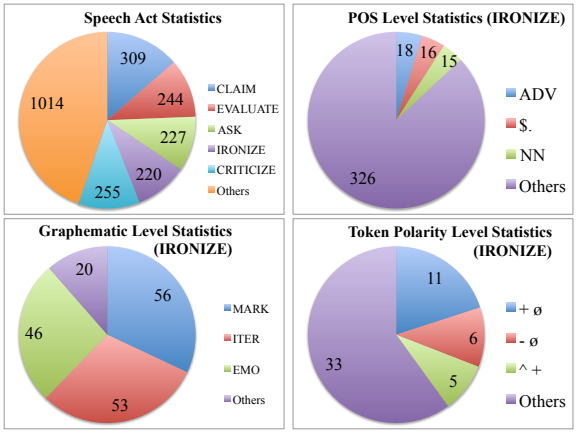


Figure 1: *HeiseTrain* corpus statistics on the target level and different annotation levels.

is ranked in the top 5 of the most often occurring speech acts in *HeiseTrain*. Second, on l_1 the most occuring tags are ADV (n=18), $. (n=16) and NN (n=15). An outstanding result is obtained for l_2: almost 90% of the most identified graphematic labels are the indicators MARK (n=56), ITER (n=53) and EMO (n=46). As most relevant patterns for token polarity, the combination of a positive token (+) and a non-valuing token (ø) are identified (n=11).

For the *HeiseTest* corpus, the multilevel annotation is carried out automatically. The POS tagging is performed by means of *WebTagger* (Neunerdt et al., 2013b) whereas level 2 and 3 as well as the basic level are annotated by means of the multilevel annotation tool *AutoAnnotator* (Trevisan et al., 2014). The *AutoAnnotator* is a rule-based and lexicon-based annotation system and uses the EXMARaLDA editor as data format. Besides POS tagging accuracies of about 95%, accuracies on other levels have to be examined in more detail.

4.2 Ironic Speech Act Patterns

Initially, multilevel patterns are determined according to the method described in 3.2 based on the *HeiseTrain* corpus. As a result of statistical evaluations, we analyze three statistical patterns with patterns over the levels l_1, l_2 and l_3. Results are depicted in the first three rows of Table1 marked as type *STAT*. The statistical pattern serve as basis for the derivation of further patterns that are modeled based on linguistic assumptions

and involve features that have been identified in previous studies, see Section 2. To be precise, we integrate the indicators l_3:(+, -) claimed by (Klenner, 2009) as well as the indicators quotation marks l_2:(*MARK*) and laughter expression l_2:(*EMO*) of (Carvalho et al., 2009). In conclusion, we obtain a type of pattern which is composed primarily of the statistical pattern and completed by additional features (type: *STAT+LING*, e.g., $P_{SL}1_{ITER}$ = P_S1_{ITER} added by l_3: "-") as well as a type of pattern that contains only linguistically motivated, non-statistical features (type: *LING*). Finally, nine patterns with features originate from two or three different levels (tuple:$|\mathbf{P}| = 2$, triple:$|\mathbf{P}| = 3$) are used and analyzed for the detection of ironic speech acts. All patterns and some *HeiseTrain* and *HeiseTest* corpus statistics are depicted in Table 1. Column five $N(M^{\mathbf{P}})$ depicts the number of exact pattern matches in the *HeiseTrain* corpus. Furthermore, the number of detected sentences with our method based on a minimum criteria fit described in 3 is given in column 6 for the gold standard corpus *HeiseTrain* (#Matches *GS*) and in column 7 for the *HeiseTest* corpus (#Matches *HT*). Finally, the occurrence of each pattern in the *HeiseTest* corpus (#Matches *HT*) is determined. The sentences with pattern matches in the *HeiseTest* corpus are extracted for pattern evaluation (see Table 2).

As evident from Table 1, the statistically determined pattern P_S2_{ITER} achieves most matches in both corpora. Rather few matches provide the linguistic patterns P_L2_{MARK} and P_L3_{MARK}.

In order to assess the usefulness of the patterns for irony detection, the extracted sentences are annotated manually and further evaluated by an inter-annotator agreement study, see Table 2. For each pattern, a set of 200 randomly chosen sentences is evaluated; less sentences are evaluated for the pattern P_L2_{MARK} and P_L3_{MARK}. Two annotators had to decide whether a sentence is an ironic or non-ironic sentence (A1 Ironic vs. A2 Ironic). Thereby, the sentence annotation is performed without considering any context, which is contrary to current methods of irony classification. For instance, (Carvalho et al., 2009) use two more classes for the annotation of unclear cases, e.g., where the context is needed or the decision. In our case, we redesigned this approach for two reasons: First, since the corpus is topic-related and the annotators are very familiar with the data, the consideration of the context can be neglected, mainly. Furthermore, giving a default class for cases, which are not clear, prevents the annotator from a clear decision, i.e., in case of doubt, the annotator would opt for the default class.

Consequently, the inter-annotator agreement between A1 and A2 is calculated (IAA(A1, A2)). In those cases, in which there is no match between A1 and A2, A3 decides whether the sentence is ironic or non-ironic (#Sentences A3). Based on the classification of the annotators, the proportion of sentences is determined that is classified by the majority as ironic. The similarities between the annotators (A1=A3; A2=A3) are listed in the last two columns (see Table 2).

The results of the inter-annotator agreement demonstrate two findings, particularly: Those patterns that brought forth the lowest number of pattern matches in Table 1 reached the best inter-annotator agreement (P_L2_{MARK} = 62.79% and P_L3_{MARK} = 63.63%, see Table 2). At the same time, the pattern that brought forth the highest number of pattern matches in Table 1 reached the lowest inter-annotator agreement (P_S2_{ITER} = 25.34%, see Table 2).

Furthermore, the inter-annotator agreement shows that the correspondence between A1 and A2 and between A2 and A3 has the largest irregularities regarding the linguistic patterns (type: LING). Here, the annotators frequently disagreed whether the examined sentence is an ironic or non-ironic sentence. In contrast, the results for the pattern of type STAT and STAT+LING are much more consistent.

5 Discussion

The results show that particularly those linguistically motivated patterns achieve a high inter-annotator agreement. The pattern with the highest inter-annotator agreement consists of self-selected, linguistic features that are based on assumptions, previous statistical results (see Section 4.1), and that are taken from the literature. However, statistical results serve as starting point for the linguistic motivation of such multilevel patterns. These results suggest two conclusions: First, the gold standard corpus used for statisti-

Pattern	Type	|P|	Patterns $M^{\mathbf{P}}$ (Tuples,Triples)	$N_I(M^{\mathbf{P}})$	#Matches *GS*	#Matches *HT*
P_S1_{ITER}	STAT	3	l_1: (\$., ADJD) l_2: (ITER) l_3: (+)	2	2	2640
P_S2_{ITER}	STAT	2	l_1: (\$., ADV, NN) l_2: (ITER)	4	17	28751
P_S3_{ITJ}	STAT	2	l_1: (\$., ITJ) l_3: (+)	2	6	3368
$P_{SL}1_{ITER}$	STAT+LING	3	l_1: (\$., ADJD) l_2: (ITER) l_3: (+, -)	0	1	421
$P_{SL}2_{ITER}$	STAT+LING	3	l_1: (\$., ADV, NN) l_2: (ITER) l_3: (+, -)	0	0	422
$P_{SL}3_{ITJ}$	STAT+LING	2	l_1: (\$., ITJ) l_3: (+, -)	1	1	549
P_L1_{MARK}	LING	3	l_1: (NN) l_2: (MARK) l_3: (+, -)	0	0	826
P_L2_{MARK}	LING	3	l_1: (ITJ) l_2: (MARK) l_3: (+, -)	0	0	43
P_L3_{MARK}	LING	2	l_2: (EMO, MARK) l_3: (+, -)	1	1	22

Table 1: Extracted patterns and their corpus frequencies in *HeiseTrain*. Explanation: P=pattern, S=statistical pattern, L=linguistic pattern, SL=statistical, linguistic pattern, ITER=iteration, MARK=marking, ITJ=interjection, P=number of pattern-inherent levels, $M^{\mathbf{P}}$=pattern, $N_I(M^{\mathbf{P}})$=exact pattern frequency in *IRONIZE* of *HeiseTrain*, #Matches *GS*=minimum citeria fit pattern frequency in *IRONIZE* of *HeiseTrain*, #Matches *HT*=minimum citeria fit pattern frequency in *HeiseTest*.

Pattern	A1 Ironic	A2 Ironic	IAA(A1,A2)	#Sent. A3	Ironic(A1,A2,A3)	A1=A3	A2=A3
P_S1_{ITER}	29.86%	35.07%	73.93%	55	30.81%	71.09%	63.98%
P_S2_{ITER}	21.72%	34.84%	66.97%	73	25.34%	73.75%	69.68%
P_S3_{ITJ}	27.96%	49.28%	64.45%	75	37.91%	64.45%	58.29%
$P_{SL}1_{ITER}$	25.82%	38.50%	71.36%	61	31.92%	68.54%	67.13%
$P_{SL}2_{ITER}$	27.11%	51.11%	65.33%	78	37.33%	62.67%	59.11%
$P_{SL}3_{ITJ}$	25.46%	47.22%	69.00%	67	33.80%	62.50%	64.81%
P_L1_{MARK}	50.95%	45.71%	70.48%	62	36.49%	53.35%	22.28%
P_L2_{MARK}	44.18%	69.77%	60.47%	17	62.79%	34.88%	51.16%
P_L3_{MARK}	59.09%	45.45%	68.18%	7	63.63%	50.00%	45.45%

Table 2: Results achieved for sample matches in *HeiseTest*. Explanation: A1=annotator 1, A2=annotator 2, A3=annotator 3, IAA=inter-annotator agreement, #Sent.A3=number of sentences annotated by A3, Ironic(A1,A2,A3)=majority decision over all annotators.

cal analysis and pattern definition with a scope of about 36,000 tokens is too small. For future studies, a larger gold standard corpus is recommended. Second, to avoid methodological effects due to the sample, the gold standard corpus, for example, should be compiled due to different selection criteria, e.g., topic or domain.

In addition, comparing the inter-annotator results with those from a previous study, it is evident that the choice of the annotators does alter the result. The annotators who conducted the inter-annotator agreement in this study are all familiar with the subject and the corpus. All three (A1, A2, A3) were involved in the development of the complete annotation scheme. However, previous studies have shown that in particular, a much higher inter-annotator agreement is reached with those annotators who had no prior knowledge regarding the annotation model or topic (Trevisan, 2014). Thus, it should be considered whether future inter-annotator agreement studies are carried out only with new, previously non-involved annotators.

With regard to the investigated pattern, other features should be taken into consideration. In the present study, only the indicators marking (label: MARK), interjection (label: ITJ) and iteration (label: ITER) are considered. A rather small proportion is ascribed to the feature emoticon (label: EMO) in contrast to the literature. Moreover, not considered features concern the semantic level and the morphological level, for example, usage regularities of topic-specific words or word types (e.g. redemptions such as *nen — einen = one*) in ironic sentences.

6 Conclusion and Outlook

In this paper, we presented a method for the automatic identification of ironic speech acts in German Web comments. As a result, ironic sentences were identified by the annotators with an accuracy of up to 63%.

Future work will focus on the iterative extraction and development of primarily linguistic patterns. To be precise, the results of the inter-annotator agreement will be validated in future

studies. Thereby, the immediate context of each sentence will be involved, i.e., the previous and the following sentence will be shown to the annotators. We assume that a higher accuracy will be achieved in the identification of irony. In addition, the investigated corpus will be enlarged in order to obtain a higher sample, identify more patterns also statistically and ensure the methods reliability.

Acknowledgments

We owe gratitude to the Excellence Initiative of the German Federal and State Government as well as Eva Reimer, Julia Ninnemann, and Simon Ruppel for their support in data processing.

References

Michael Beißwenger, Maria Ermakova, Alexander Geyken, Lothar Lemnitzer, and Angelika Storrer. 2012. A TEI Schema for the Representation of Computer-mediated Communication. *Journal of the Text Encoding Initiative*, pages 1 – 31.

Ulla Bohnes. 1997. Compas-b. beschreibung eines forschungsprojektes. magisterarbeit im fach neuere deutsche sprachwissenschaft. Master's thesis, Universität des Saarlandes.

Paula Carvalho, Luís Sarmento, Mário J. Silva, and Eugénio de Oliveira. 2009. Clues for Detecting Irony in User-Generated Contents: Oh...!! It's "So Easy" ;-). In *Proceedings of the 1st International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion*, TSA '09, pages 53–56, New York, NY, USA. ACM.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 42–47.

Manfred Klenner. 2009. Süsse Beklommenheit und schmerzvolle Ekstase. Automatische Sentimentanalyse in den Werken von Eduard von Keyserling. *Tagungsband der GSCL-Tagung, Gesellschaft für Sprachtechnologie und Computerlinguistik*, 30(2).

Rada Mihalcea and Carlo Strapparava. 2006. Learning to Laugh (automatically): Computational Models for Humor Recognition. *Computational Intelligence*, 22(2):126–142.

Melanie Neunerdt, Bianka Trevisan, Rudolf Mathar, and Eva-Maria Jakobs. 2012. Detecting Irregularities in Blog Comment Language Affecting POS Tagging Accuracy. *International Journal of Computational Linguistics and Applications*, 3(1):71–88, June.

Melanie Neunerdt, Michael Reyer, and Rudolf Mathar. 2013a. A POS Tagger for Social Media Texts trained on Web Comments. *Polibits*, 48:59–66.

Melanie Neunerdt, Bianka Trevisan, Michael Reyer, and Rudolf Mathar. 2013b. Part-of-Speech Tagging for Social Media Texts. In *International Conference of the German Society for Computational Linguistics and Language Technology (GSCL)*, pages 139–150, Darmstadt, Germany, September.

Martha Ripfel. 1987. Was heißt bewerten? *Deutsche Sprache*, 15:151–177.

Barbara Sandig. 1979. Ausdrucksmöglichkeiten des bewertens. ein beschreibungsrahmen im zusammenhang eines fiktionalen textes. *Deutsche Sprache*, 7:137–159.

Barbara Sandig. 2006. *Textstilistik des Deutschen*. de Gruyter, Berlin/New York.

Anne Schiller, Simone Teufel, Christine Stöckert, and Christine Thielen. 1999. Guidelines für das Tagging deutscher Textcorpora mit STTS. University of Stuttgart.

Helmut Schmid. 1995. Improvements in part-of-speech tagging with an application to German. In *In Proceedings of the ACL SIGDAT-Workshop*. Citeseer.

Bianka Trevisan and Eva-Maria Jakobs. 2010. Talking about mobile communication systems: verbal comments in the web as a source for acceptance research in large-scale technologies. In *Professional Communication Conference (IPCC), 2010 IEEE International*, pages 93–100.

Bianka Trevisan, Melanie Neunerdt, and Eva-Maria Jakobs. 2012. A Multi-level Annotation Model for Fine-grained Opinion Detection in German Blog Comments. In *11th Conference on Natural Language Processing (KONVENS)*, pages 179–188, Vienna, Austria, September.

Bianka Trevisan, Tim Hemig, and Eva-Maria Jakobs. 2014. *AutoAnnotator: A Tool for Automated Multilevel Annotation of Web Comments*. In preparation.

Bianka Trevisan. 2014. *Bewerten in Blogkommentaren. Mehrebenenannoation sprachlichen Bewertens*. RWTH Aachen University.

Mining corpora of computer-mediated communication: Analysis of linguistic features in Wikipedia talk pages using machine learning methods

Michael Beißwenger (TU Dortmund)
Harald Lüngen (IDS Mannheim)
Eliza Margaretha (IDS Mannheim)
Christian Pölitz (TU Dortmund)

Abstract

Machine learning methods offer a great potential to automatically investigate large amounts of data in the humanities. Our contribution to the workshop reports about ongoing work in the BMBF project KobRA (http://www.kobra.tu-dortmund.de) where we apply machine learning methods to the analysis of big corpora in language-focused research of computer-mediated communication (CMC). At the workshop, we will discuss first results from training a Support Vector Machine (SVM) for the classification of selected linguistic features in talk pages of the German Wikipedia corpus in DeReKo provided by the IDS Mannheim. We will investigate different representations of the data to integrate complex syntactic and semantic information for the SVM. The results shall foster both corpus-based research of CMC and the annotation of linguistic features in CMC corpora.[1]

1 Introduction

Up to now there have been very few annotated corpora of CMC freely available for the scientific community. Scholars doing data-based research of CMC discourse therefore often face the following limitations:

(a) They have to collect corpora for their research projects by themselves.

(b) "Off the shelf" tools for the linguistic annotation of written language data do not perform on CMC data in a satisfying way.

(c) Given (a) and (b), the researchers either have to annotate their corpora manually or confine themselves to analyzing their corpora as raw data (without the possibility to query linguistic annotations).

(d) The corpora they are able to analyze (taking into consideration that (a) and (c) are consuming a lot of their time and effort) are rather small than big.

The methods and experiments described in this paper are driven by the vision that the application of machine learning methods can improve the situation and possibilities of building corpora and doing corpus-based analysis of CMC discourse in several respects:

1. If we succeed to adapt machine learning methods for the automatization of typical routine tasks in corpus-based analysis (e.g. the cleaning and classification of query results), then these methods can support linguists in analyzing bigger data than they could analyze when every routine task would have to be done manually. "Big data", here, refers to amounts of data which are too large to be analyzed intellectually. For a linguist, the Wikipedia which is used as the test bed for the experiments reported here definitely *is* "big data": The German Wikipedia corpus in DeReKo comprises more than 1.5 million article pages (consisting of 678 million word tokens) and more than 555,000 talk pages (consisting of 264 million word tokens).

2. The methods applied can be used not only for mining the big data for those "gold nuggets" which are relevant for a particular linguistic research question; they may additionally be

used as a basis for automatically annotating the retrieval and classification results. In this respect, machine learning methods also enhance the conditions for building annotated CMC corpora.

In the following sections we give an overview of the project background of our work (sect. 2), a description of the Wikipedia corpus in DeReKo (sect. 3), and a description of the linguistic phenomena under observation (sect. 4). Sect. 5 describes the machine learning methods applied and sect. 6 gives an outlook on ongoing and future work.

2 Project background

The work presented in our paper is part of the KobRA project ("Corpus-based linguistic research and analysis using data mining") funded by the eHumanities program of the BMBF 2012-2015.[2] The project brings together researchers from linguistics, language technology and artificial intelligence to adapt machine learning methods for recurrent and time-consuming routine tasks that linguists have to perform when doing corpus-based linguistic analysis (e.g. classification and disambiguation of results from corpus queries) and thus to enable researchers to work with amounts of data that are too big to be be analyzed intellectually. The application scenario for the methods developed in the project is defined in case studies from several fields of linguistic research: diachronic linguistics, lexicography, variational linguistics/computer-mediated communication.

The data basis and test bed for the experiments reported in this paper is the German Wikipedia corpus in DeReKo provided by the IDS Mannheim (cf. sect. 3) on which the methods are trained and evaluated and which allows for a comparison of language use in monologic texts (= "article pages") and in dialogic written conversations (the sequences of user postings that can be found on "talk pages") which, cum grano salis, are both usually written by the same user group (= those users who contribute to writing Wikipedia articles as authors, moderators, reviewers etc.). Previous research has shown that Wikipedia is a fruitful resource for studies in linguistic variation on the internet (Storrer, 2013).

The scope of the experiments is on the retrieval and automatic classification of selected linguistic phenomena which can be considered as either specific for language use in written CMC or as elements which are typical of language use under the conditions of spontaneous, dialogic interaction and which occur both in spoken conversations as well as in written conversations on the internet (cf. sect. 4).

3 The corpus

The CMC corpus we used for the experiments is the 2013 conversion of the Wikipedia available within DeReKo, the German Reference Corpus (Kupietz and Lüngen, 2014), at the *Institut für Deutsche Sprache* in Mannheim.[3] It was built from the Wikipedia dump of July 27, 2013, and contains approx. 943 million tokens. Unlike other corpora derived from Wikipedia, it has been prepared as a linguistic corpus and comprises the whole German Wikipedia. It is represented in I5 (Lüngen and Sperberg-McQueen, 2012) the TEI P5 customization used to encode the texts in DeReKo.

Since the Wikipedia talk pages corpus was one of the first sub-corpora in DeReKo to contain CMC texts, the I5 format was on this occasion extended to incorporate macro-structural elements (most notably <posting>) and attributes to represent the thread and posting structure of CMC data as proposed in (Beißwenger et al., 2012).

In Wikipedia, each talk page (or: discussion) is paired with a Wikipedia article. On a talk page, the users, i.e. Wikipedia authors, can discuss an article, i.e. whether and how it should be revised or extended, what references or images to include etc. When an article is edited, the editor usually justifies his/her edit by a written contribution on the respective talk page. According to the Wikipedia talk page guidelines[4] and also in prac-

[2] See http://www.kobra.tu-dortmund.de. The project is headed by Angelika Storrer (U Mannheim/German Linguistics. The main partners of the project are Katharina Morik (TU Dortmund University/Artificial Intelligence), the IDS Mannheim (Marc Kupietz, Andreas Witt), the BBAW Berlin (Alexander Geyken) and the SfS at U Tübingen (Erhard Hinrichs/Computational Linguistics).

[3] see http://www.ids-mannheim.de/kl/projekte/korpora/verfuegbarkeit.html

[4] http://de.wikipedia.org/wiki/Wikipedia:Diskussionsseiten

tice, a talk page is structured much like a discussion forum, i.e. it comprises a sequence of discussion topics introduced by headings, and within such a topic, dialogue turn(Schegloff, 2007)-like units provided by a single user are delimited by means of paragraph indentation, thus forming a discussion thread. (Beißwenger et al., 2012) classify these turn-like units as *posting* units, and this view has also been adopted in the I5 representation of the Wikipedia corpus in DeReKo.[5]

The conversion of the wikitext data of the Wikipedia dump into the I5 format is described in detail in (Margaretha and Lüngen, In press), the source code of the conversion tools is available from GitHub.[6] The conversion pipeline also includes a heuristic method for identifying the posting segments in a talk page and an evaluation of this method. According to the evaluation on 49 talk pages, the performance of the automatic heuristic posting segmentation yielded approximately 60% micro average precision and 80% micro average recall when compared with posting segmentations provided by human annotators. The agreement between the two human annotators themselves was κ=0.76, which suggests that the exact identification of posting boundaries is not an unambiguous tasks for humans, either, when reading a talk page. Altogether 5.4 million posting segments were identified and annotated in the talk pages corpus by the automatic segmentation. For the corpus, PoS annotations from the Stuttgart TreeTagger are also available (though they have not been used in the experiments described here), and we have prepared Wikipedia corpora in the same fashion for other languages, too.

[5]A posting in CMC is originally defined as a piece of text sent to the server by the author at one specific point in time. Hence, the turn-like sections in Wikipedia talk pages are strictly speaking not postings, as a wiki user always posts a new version of the whole wiki page, i.e. (s)he might have edited the page in different places, even might have modified or deleted previous contributions by other users. But since on a talk page, the dialogue structure with its sequentially ordered threads of turns prevails, the turn-like units have been identified with postings as defined in (Beißwenger et al., 2012) in the present I5 representation.

[6]https://github.com/IDS-Mannheim/Wikipedia-Corpus-Converter

4 Machine learning tasks

For our first experiments with adapting machine learning methods for the analysis and annotation of Wikipedia, we selected two types of linguistic features which are of interest for studies in language-focused CMC research as well as for research on linguistic variation in written and spoken language.

4.1 Interaction words

Interaction words are units which are based on a word or a phrase of a given language describing expressions, gestures, bodily actions, or virtual events. In German CMC, simple forms of interaction words typically have the form of non-inflected verb stems (*grins, lach, freu*) whereas complex forms additionally may incorporate objects and/or adverbials (*lautlach*, *diabolischgrins, kopfschüttel, schulterzuck, nachlinksrutsch*). Some interaction words have the form of acronyms (*lol*, *rofl*, *g*). Interaction words are usually not part of the syntactic structure of the utterance they accompany; instead, they are used for the description of emotions or mental activity, as illocution or irony markers, or to playfully mimic bodily activity (Beißwenger et al., 2012). They are often (but not necessarily) enclosed in asterisks (**grins**, **freu**).

As a starting point for our experiments in automatically detecting interaction words, we assume that a researcher who wants to analyze interaction words in a corpus where these units are not explicitly annotated would usually define a query pattern for expressions which s/he considers as typical forms of interaction words – for example forms which are frequently used as interaction words in other corpora or random expressions between asterisks. We defined tasks for both of these two scenarios:

Task #1a:

- *Data basis:* Query results for the most frequent forms of interaction words according to the annotations in the Dortmund Chat Corpus (*lol, lach, freu, grins, wink, seufz*; cf. (Storrer, 2013). Each match is represented in a snippet with a context size of max. 999 characters (extracted from the corpus).

- *Training and evaluation data:* Random sample with 600 matches from the data basis that have been independently classified by two human annotators as "contains an interaction word" (type 1) or "does not contain an interaction word" (type 0).
- *Task:* Learn a classification model for separating the snippets into type 1 and type 0 snippets.

Task #1b:

- *Data basis:* Query results for expressions between asterisks. Each match is represented in a snippet with a context size of max. 999 characters (extracted from the corpus).
- *Training and evaluation data:* Random sample with 600 matches from the data basis that have been independently classified by two human annotators as "contains an interaction word" (type 1) or "does not contain an interaction word" (type 0).
- *Task:* Learn a classification model for separating the snippets into type 1 and type 0 snippets.

4.2 "Non-canonical" uses of *weil* and *obwohl*

In the written German standard, *weil* and *obwohl* are conjunctions which introduce subordinate clauses with the finite verb form in sentence-final position. Under conditions of conceptual orality (prototypically but not limited to spontaneously spoken language), *weil* and *obwohl* also occur in the pre-front position of sentences with the finite verb in a position other than sentence-final (typically V2; examples: *"ja toll aber so richtig steht es nicht drin* ***weil*** *damals sollten wir nämlich eine arbeit in informatik machen über das dualsystem", "Ja ich bin auch 96 Fan aber trotzdem, er hätte auch im Spiel sein fehler noch ändern können.* ***Weil*** *ich bin selber Schiedsrichter, und hatte auch schon so eine Situation"*). In popular discussions about language change, cases like these are often considered as degenerated grammar and as an example of language decline (cf. critically on this discussion: (Günthner, 2008) while analysis in the field of spoken language research/interactional linguistics could show that in their "non-canonical" uses *weil* and *obwohl* often have functions which are different from those of the "canonical" use as subordinate conjunctions (cf. e.g. (Gohl and Günthner, 1999), (Günthner and Auer, 2005), (Imo, 2012). It is an open question inhowfar "non-canonical" uses of *weil* and *obwohl* in written CMC have the same or similar functions as "non-canonical" uses in spoken language. Corpus-based analyses on this question will help to develop a better understanding of how much the encoding medium (writing vs. articulated sound) and the structure of the encoding process (private composition before transmission vs. 'on-line') affect the structure of utterances in written and spoken conversations.[7]

Our first experiments addressed the classification of matches for *weil* in the corpus:

Task #2:

- *Data basis:* All 305,708 matches for *weil* in the talk pages subcorpus. Each match is represented in a snippet with a context size of max. 999 characters (extracted from the corpus).
- *Training and evaluation data:* Random sample with 1,200 matches from the data basis that have been independently classified by two human annotators as "non-canonical use" (type 1) or as "canonical use" (type 0).
- *Task:* Learn a classification model for separating the snippets into type 1 and type 0 snippets.

5 Machine learning methods

Machine learning methods offer automatic classification and filter methods for large scale data. Based on examples, a decision function is extracted that can be applied to large amounts of data to classify and filter them with respect to the CMC phenomena like those described in section 4. The collection of all these extracted rules is summarized by a single classification model. The derivation of such rules depends on the features of the

[7] Cf. the discussion of the effect of written 'en bloc' encoding on the process of message composition and the system of turn-taking in (Beißwenger, 2007)

data as well as on the complexity and regularities in the texts.

We use kernel methods (Shawe-Taylor and Cristianini, 2004) and Support Vector Machines to integrate different feature representations of the corpus snippets into a classification model. A Kernel encodes similarity information for pairs of snippets based on a certain feature representation. Kernel methods enable us to directly integrate all possible feature representations of the data – even complex representations such as syntactic structures or semantic relations – into a single classification model. This model is a Support Vector Machine that uses the Kernels to decide which snippets belong to a certain class and which not.

We use three different kernels to represent the snippets from the Wikipedia corpus: A *tree kernel* is used to integrate syntactic information from parse trees as proposed by (Moschitti, 2006). To derive the parse trees for German sentences, we use the Stanford Parser (Rafferty and Manning, 2008). Further information is integrated via *Substring kernels* that count the presence of certain substrings in a given text (Lodhi et al., 2002). Last, a linear kernel is used on the *bag-of-words* representations of the corpus snippets. In the bag-of-words representation, each snippet is represented via a large vector. Each component of such a vector gives the (normalized) frequency of a certain word appearing in the text. This is the baseline approach which we compare to the kernel methods.

In order to use the kernels for the classification of the phenomena under observation, we generate a Gram matrix for each of them. The Gram matrix contains the kernel evaluations for each pair of snippets from the training data. These evaluations are everything needed to learn our classification model.

For each Gram matrix, we train a Support Vector Machine using the LibSVM library (Chang and Lin, 2011). The Support Vector Machine uses the Gram matrix to learn a decision function that is used to classify any snippet for the respective phenomena. For both the training of the classification model and its application on test data, we only use kernel evaluations from the Gram matrix.

The training is done on a part of the hand-classified training data described in section 4. Then we apply the Support Vector Machine to the rest of the data to classify them for the phenomenon. Based on this independent test set, the performance of the classifier can be evaluated and we can estimate which kernel is best suited for the task.

In order to estimate the performance, we perform a 10-fold cross validation evaluation. The measure of the performance is the F1 score, that is the mean of the precision and the recall of the trained classifier. Finally, the model is applied to the unlabeled test data. In order to get information on what snippets are difficult to classify, we additionally estimate confidence values of the classification. These values are used to propose additional hand classifications for some of the snippets. In an *Active Learning* (Settles, 2009) setting, this potentially results in better training data by actively choosing which snippets to classify by hand.

6 State of work and future agenda

At the KONVENS workshop, we will present and discuss first results from adapting the machine learning methods outlined in sect. 5 for the retrieval and disambiguation tasks described in sect. 4. As next steps, we are planning to further improve these results by using additional methods (Active Sampling), by doing experiments with different data sets for the same phenomena and by adapting the models which perform well also to data sets from other CMC genres/corpora.

The optimized classification models shall finally be used for automatically annotating the results in the corpus data. For this purpose, we will use labels from the extended STTS tagset for the POS tagging of CMC corpora ("STTS-IBK") that has been defined for the Empirikom shared task on linguistic processing of German CMC (*EmpiriST2015*[8]).

As a part of our future agenda, we are planning to transfer the machine learning methods described in this paper also to other genres and phenomena: On the one hand, the classifiers trained on Wikipedia talk pages shall be evaluated with/adapted to data also from Wikipedia articles pages and from other CMC genres such as chats, tweets, or blog comments. On the other

[8] http://empirikom.net/bin/view/Themen/SharedTask

hand, the methods developed for the identification/classification of interaction words and "non-canonical" *weil/obwohl* shall be adapted also to other linguistic phenomena which are of interest for language-focused corpus investigations of CMC discourse. In this context, we will also investigate which approaches for text representations in the field of machine learning are important to safely apply our trained models to new and unseen texts and phenomena, and examine and compare our methods to previous domain adaptation methods like FLORS (Schnabel and Schuetze, 2014).

References

Michael Beißwenger, Maria Ermakova, Alexander Geyken, Lothar Lemnitzer, and Angelika Storrer. 2012. A TEI schema for the representation of computer-mediated communication. *Journal of the Text Encoding Initiative*, 3.

Michael Beißwenger. 2007. *Sprachhandlungskoordination in der Chat-Kommunikation*, volume 26 of *Linguistik – Impulse & Tendenzen*. de Gruyter, Berlin. New York.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. Technical report, ACM Transactions on Intelligent Systems and Technology. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm .

Christine Gohl and Susanne Günthner. 1999. Grammatikalisierung von *weil* als Diskursmarker in der gesprochenen Sprache. *Zeitschrift für Sprachwissenschaft*, 18(12):39–75.

Susanne Günthner and Peter Auer. 2005. Die entstehung von diskursmarkern im deutschen – ein fall von grammatikalisierung? In Torsten Leuschner, Tanja Mortelsmans, and Sarah de Groodt, editors, *Grammatikalisierung im Deutschen*, pages 335–362. de Gruyter, Berlin.

Susanne Günthner. 2008. Geht die nebensatzstellung im deutschen verloren? In Markus Denkler, Susanne Günthner, Wolfgang Imo, Jürgen Macha, Dorothee Meer, Benjamin Stoltenburg, and Elvira Topalovicet, editors, *Frischwärts und unkaputtbar. Sprachverfall oder Sprachwandel im Deutschen*, pages 103–128. Aschendorff, Münster.

Wolfgang Imo. 2012. Wortart diskursmarker? In Björn Rothstein, editor, *Nicht-flektierende Wortarten*, pages 48–88. de Gruyter, Berlin.

Marc Kupietz and Harald Lüngen. 2014. Recent developments in dereko. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini, and Chris Watkins. 2002. Text classification using string kernels. *Journal of Machine Learning Research*, 2:419–444, March.

Harald Lüngen and Michael Sperberg-McQueen. 2012. A TEI P5 Document Grammar for the IDS Text Model. *Journal of the Text Encoding Initiative*, 3:1–18.

Eliza Margaretha and Harald Lüngen. In press. Building linguistic corpora from wikipedia articles and discussions. *Journal for Language Technology and Computational Linguistics (JLCL)*.

Alessandro Moschitti. 2006. Making tree kernels practical for natural language learning. In Diana McCarthy and Shuly Wintner, editors, *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006), April 3-7, 2006, Trento, Italy*, pages 113–120. The Association for Computer Linguistics.

Anna Rafferty and Christopher D. Manning. 2008. Parsing Three German Treebanks: Lexicalized and Unlexicalized Baselines. In *Proceedings of the ACL Workshop on Parsing German*.

Emanuel Schegloff. 2007. *Sequence Organization in Interaction*, volume 1: A Primer in Conversation Analysis. Cambridge University Press, UK.

Tobias Schnabel and Hinrich Schuetze. 2014. Flors: Fast and simple domain adaptation for part-of-speech tagging. In *Transactions of Association for Computer Linguistics*, pages 15–26.

Burr Settles. 2009. Active learning literature survey. Technical Report 1648, University of Wisconsin–Madison. Computer Sciences Technical Report.

John Shawe-Taylor and Neollo Cristianini. 2004. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA.

Angelika Storrer. 2013. Sprachstil und Sprachvariation in sozialen Netzwerken. In Barbara Frank-Job, Alexander Mehler, and Tilmann Sutter, editors, *Die Dynamik sozialer und sprachlicher Netzwerke. Konzepte, Methoden und empirische Untersuchungen an Beispielen des WWW*, pages 331–366. VS Verlag für Sozialwissenschaften, Wiesbaden.

Network of the Day: Aggregating and Visualizing Entity Networks from Online Sources*

Darina Benikova, Uli Fahrer, Alexander Gabriel, Manuel Kaufmann,
Seid Muhie Yimam, Tatiana von Landesberger, Chris Biemann
Computer Science Department, TU Darmstadt, Germany
www.tagesnetzwerk.de

Abstract

This software demonstration paper presents a project on the interactive visualization of social media data. The data presentation fuses German Twitter data and a social relation network extracted from German online news. Such fusion allows for comparative analysis of the two types of media. Our system will additionally enable users to explore relationships between named entities, and to investigate events as they develop over time. Cooperative tagging of relationships is enabled through the active involvement of users. The system is available online for a broad user audience.

1 Introduction

The constantly growing interest in social media raises a need for new tools enabling wide audience to analyze and explore the available data. Our work addresses this need via the interactive online visual system *Network of the Day* (Netzwerk des Tages). It combines information extracted from the social media platform Twitter and online newspaper articles. *Network of the Day* offers a transparent exploration of current media to politically interested non-experts.

The visualization shows the most important current entities discussed in online media in a compact and interactive form. The presented data is kept up to date on a daily basis. We present the media data in several interlinked views. First, we extract and show the relationships between entities (i.e., persons and organizations) in a network. Interaction with this network enables the users to tag the relations between entities, which creates additional semantics in the data. Second, a line chart shows the occurrences of most popular entities for the respective day over the past months. This offers the possibility to spot the development of important topics over time. Third, this enables the user to compare commonalities and differences of the two media. Finally, the user can search for entities of her interest in order to gain information on media developments, which are of relevance to her.

2 Related work

Summarizing and extracting information from media databases has been a task of great interest in natural language processing, as the amount of information is too large to be processed by humans without automatic aids.

In recent years, the possibilities of opinion expression or social-media communication have increased, resulting in a surge of sentiment analysis tools (Pang and Lee, 2008). Especially there is a need for filtering and exploring events and opinions in high-volume social media data.

The visualization of social network data, Twitter data and news has gained importance. Several approaches have been developed. TextViz[1] provides an overview of text visualization techniques from various areas. Most relevant to our work are the visualization of word co-occurrence in Twit-

[1] http://textvis.lnu.se

ter messages and visualizations of relations between named entities. For example, Phrase Nets (Van Ham et al., 2009) show co-occurrence of words as a network, however they do not allow for exploring time dependent changes. On the contrary, Topic Competition (Xu et al., 2013) shows the development of word and topic frequencies over time. However, the relationships between topics and entities are not visible. A further relevant work by Biemann et al. (2004) shows paths through networks extracted from news. While this software is interactive, relations between entities cannot be labeled interactively and developments over time are not shown.

In this work, the social media communication is represented by the Twitter[2] platform. Meckel and Stanoevska-Slabeva (2009) investigated the reflexion of politics upon Twitter. *Twitterbarometer* [3] is a tool developed by the Buzzrank company which measures the political mood in real time by capturing tweets related to parties – as indicated by hashtags – and classifying them as positive or negative.

3 Description of main components

This section presents the main components of the project. We first describe the data sources, their deployment and their processing. We then present two main components of the project – the *Twitter contrast analysis* and *Network of Names*. These components form a basis of the new system presented in Section 4.

3.1 Data Sources

The data sources used in our system are online news from "Wörter des Tages" and online messages from Twitter.

3.1.1 Online News

The project "Wörter des Tages"[4] (Quasthoff et al., 2002) serves as our source of daily news articles. Frequently appearing words are extracted daily by a text mining suite from daily newspapers and news services.

The project "Wörter des Tages" extracts its data mostly from German online sites, resulting in a daily dataload of approximately 20,000 - 50,000 sentences. The texts are segmented and indexed, the terms are quantitatively acquired and statistically significant co-occurrences are computed. The main parameters for the term selection are the frequency in the current daily corpus, the frequency in the already mentioned reference corpus "Deutscher Wortschatz" and the factor of relative frequencies between the two corpora of the term (Quasthoff et al., 2002).

3.1.2 Twitter

We download Twitter data using its public Streaming API[5] that gives developers access to Twitter's global stream of Tweet data. This stream is filtered according to previous selected most important keywords, i.e. as extracted by (Quasthoff et al., 2002).

3.2 Basis Software Components

Two recent works form the basis of this project: Fahrer's implementation (2014) of a Twitter contrast-analysis, which shows words frequently co-occurring with search terms and the work of Kochtchi et al. (2014), which visualizes the relationships between people and organizations using online newspaper articles as a source. Both projects provide full provenance information, i.e. users are not only able to see and manipulate the display of automatically extracted relationships, but also to access the text sources from which the relationships are extracted.

3.2.1 Twitter contrast-analysis

The component by Fahrer (2014) provides a contrastive co-occurrence analysis that contrasts two separate keywords regarding their strongly associated words in Twitter messages. For example, Figure 1 shows a contrastive analysis for the keywords *Brüderle* and *Trittin*, who are prominent German politicians from two different parties. The left side of the graph shows words only co-occurring with the keyword Brüderle and the right side shows only co-occurring words with Trittin. The overlap in the middle indicates words that are co-occurring with both terms. Results

[2] http://www.twitter.com

[3] http://twitterbarometer.de

[4] http://www.wortschatz.uni-leipzig.de/wort-des-tages/

[5] https://dev.twitter.com/docs/api/

show that the overlap in the contrast analysis gives a sensible reflection of main political events. Furthermore, most of the relevant newspaper topics regarding the contrastive analysis are reflected in Twitter.

The data for a study on the German parliament election was collected from Twitter between August 2, 2013 and October 9, 2013. Overall a corpus of 10,524,367 Twitter messages was collected. For the tokenization, the Twitter tokenizer from Gimpel et al. (2011) was employed. To determine the words strongly co-occurring with a given word the log-likelihood measure (Dunning, 1993) was applied to rank the vocabulary according to descending values (Fahrer, 2014).

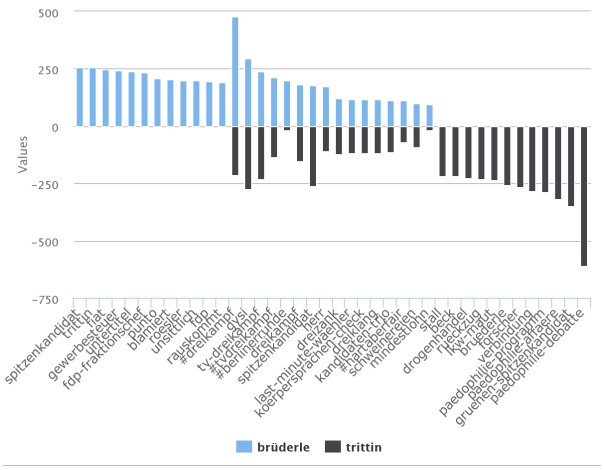


Figure 1: Sample contrastive analysis with the search terms “Brüderle” (light bars) and “Trittin” (dark bars) with 40 result terms, cf. (Fahrer, 2014)

3.2.2 Network of Names

The second basic component is the exploration of relationships between named entities presented by Kochtchi et al. (2014). This interactive system derives a social network graph from information extracted from online publications of newspaper articles.

The visualization enables to explore and investigate the relationships between people and organizations of public interest, reflecting the interaction between public protagonists and the influence of their surroundings, sociality and public policy. Kochtchi et al. (2014) used the Leipzig Corpora Collection (Richter et al., 2006), containing about 70 million of sentences extracted from German online newspapers between 1995 and 2010, as the text source of his project. In the course of preprocessing, Kochtchi et al. (2014) extracted Named Entities using the Stanford Named Entity Parser (Faruqui and Padó, 2010; Finkel et al., 2005) and calculated normalized PMI scores (Bouma, 2009) of co-occurrence. The Network of Names component offers the possibility of collaborative social tagging. By clicking on the edges between entities, users can enter a relation label of this relationship. The users base these labels on the sentences containing the two entities. The sentences are shown in an extra frame next to the relationship. While the Network of Names was a static visualization of a large corpus, we use parts of this technology to create daily networks and components display changes over time.

4 Combination of social-media and computer-mediated communication

The main goal of “Network of the Day” is to present current main topics and their relationship on the basis of combining *online news* and *social media*. The combination represents the contrast of the presentation of events by the German online media and the reaction to the situation of a part of the German online Twitter community.

Figure 2 illustrates the visualization for networks extracted from daily news. Our visualization comprises four main parts, which are interactively linked: daily network, social tagging, time line and twitter contrast analysis.

Networks are constructed on a daily basis, represent important events of the day, and can be visually compared to networks from the past. Each network shows the relationships between the most important persons and organisations of the day. Entities are nodes and their co-occurrence is denoted by edges. The user can select entities from the graph and their most important co-occurring terms over time. The network is clustered with the Markov Cluster Algorithm (van Dongen, 2000), and clusters can be unfolded and collapsed by clicking on them. Cluster labels are the most central three nodes within a cluster that are calculated using the Pagerank algorithm (Page et al., 1999). We use a flexible force-directed layout for

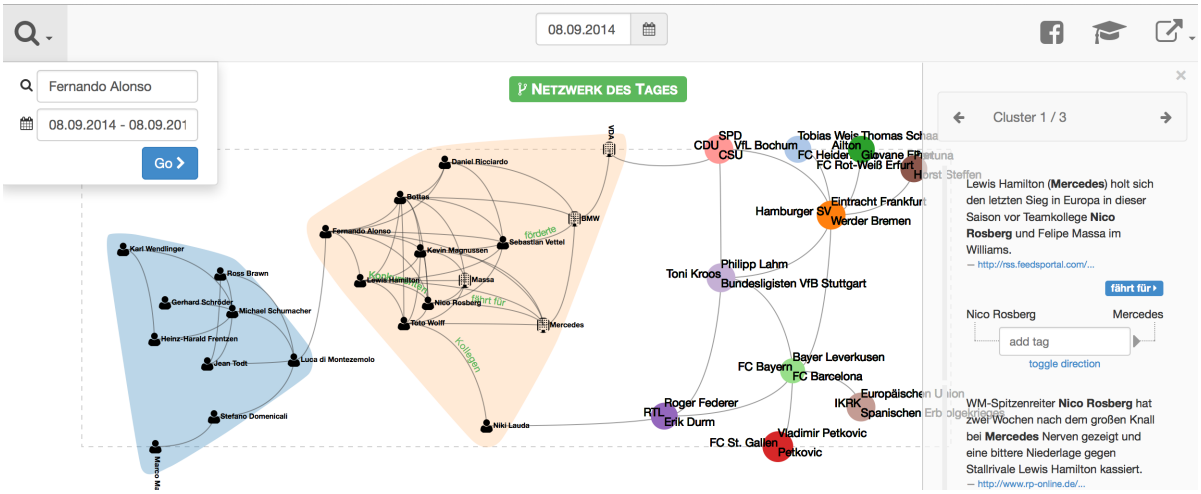


Figure 2: Visualization of a Network of the Day for September 8, 2014 after a search for ”Fernando Alonso”. Two clusters about motor sports are unfolded, the sources for the link between ”Nico Rosberg” and ”Mercedes” are shown and their relation is labeled as ”fährt für” (drives for).

the graph rendering that is implemented using the D3.js[6] JavaScript visualization library.

Clicks on links result in the display of source sentences, which are linked to the original online articles. Users can tag relationships of entities using the *interactive social tagging component*, see right side of Figure 2. Further, selecting an edge also invokes a contrast analysis of the two connected entities based on Twitter data, cf. Section 3.2.1 (not shown due to space constraints). The search mask allows the user to search for entities of her choice in arbitrary time spans, and to obtain a detailed analysis. This allows for user specific exploration of current and past social media.

The dynamics of word frequency over time is exemplified in Fig. 3 and displayed below the network. Initially, it shows terms that were popular on the respective day, but arbitrary terms from the network can be selected, and compared in the frequency diagram.

5 Outlook and Further work

Network of the Day offers a transparent aggregation of current media to laymen interested in politics and other daily affairs. Moreover, it offers them the possibility to collaboratively tag interesting relationships. Very importantly, the visualization provides full provenance, as original sources are linked.

[6] http://d3js.org/

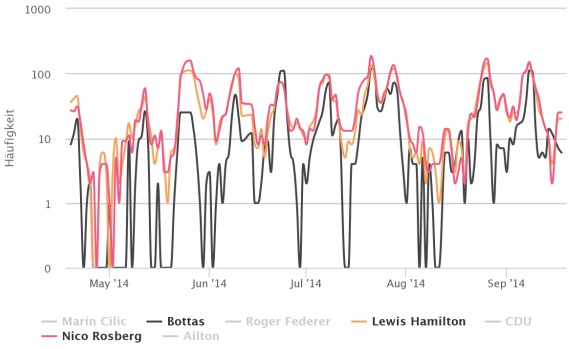


Figure 3: Frequency diagram of trending terms on September 8, 2014, reflecting the bi-weekly schedule of Formula 1 races.

By extracting the current information on relations, people, organizations and events from Twitter, the result of this project may be used in political education or serve voters as an overview. In this study only a comparison of data containing the search terms, as described above, may be provided. In a further study, a direct comparison of entities such as persons, organizations and events, appearing in both Twitter and online newspaper articles may be conducted.

The software is available as an online website[7], and is expected to be finalized in October 2014.

[7] available on http://maggie.lt.informatik.tu-darmstadt.de/nod/ via http://tagesnetzwerk.de/

Acknowledgements

"Netzwerk des Tages" (Network of the Day) is funded by BMBF via a grant from Hochschulwettbewerb 2014[8].

References

Chris Biemann, Karsten Böhm, Gerhard Heyer, and Ronny Melz. 2004. Automatically building concept structures and displaying concept trails for the use in brainstorming sessions and content management systems. In *Proceedings of I2CS*, Guadalajara, Mexico. Springer LNCS.

Gerlof Bouma. 2009. Normalized (Pointwise) Mutual Information in Collocation Extraction. In *Proceedings of the International Conference of German Society for Computational Linguistics and Language Technology*, pages 31–40, Potsdam, Germany.

Ted Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational linguistics*, 19(1):61–74.

Uli Fahrer. 2014. Contrastive Co-occurrence Analysis on Twitter for the German Election 2013. In *GI-Edition: Lecture Notes in Informatics*, pages 257–260, Potsdam, Germany.

Manaal Faruqui and Sebastian Padó. 2010. Training and evaluating a German named entity recognizer with semantic generalization. In *Proceedings of KONVENS*, pages 129–133, Saarbrücken, Germany.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370, Michigan, USA.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech Tagging for Twitter: Annotation, Features, and Experiments. In *Proc. ACL-HLT-2011*, pages 42–47, Portland, OR, USA.

Artjom Kochtchi, Tatiana von Landersberger, and Chris Biemann. 2014. Networks of Names: Visual Exploration and Semi-Automatic Tagging of Social Networks from Newspaper Articles. *Computer Graphics Forum*, 33(3).

Miriam Meckel and Katarina Stanoevska-Slabeva. 2009. Auch Zwitschern muss man üben: Wie Politiker im deutschen Bundestagswahlkampf "twitterten". *Neue Zürcher Zeitung*.

[8] `http://www.hochschulwettbewerb2014.de/`

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November. Previous number = SIDL-WP-1999-0120.

Bo Pang and Lillian Lee. 2008. Opinion Mining and Sentiment Analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135.

Uwe Quasthoff, Matthias Richter, and Christian Wolff. 2002. " Wörter des Tages"-Tagesaktuelle wissensbasierte Analyse und Visualisierung von Zeitungen und Newsdiensten. In *ISI*, pages 369–372.

Matthias Richter, Uwe Quasthoff, Erla Hallsteinsdóttir, and Chris Biemann. 2006. Exploiting the Leipzig Corpora Collection. In *Proceedings of the IS-LTC 2006*, pages 68–73, Ljubljana, Slovenia.

Stijn van Dongen. 2000. *Graph Clustering by Flow Simulation*. Ph.D. thesis, University of Utrecht, Utrecht.

Frank Van Ham, Martin Wattenberg, and Fernanda B Viégas. 2009. Mapping text with phrase nets. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):1169–1176.

Panpan Xu, Yingcai Wu, Enxun Wei, Tai-Quan Peng, Shixia Liu, Jonathan JH Zhu, and Huamin Qu. 2013. Visual analysis of topic competition on social media. *Visualization and Computer Graphics, IEEE Transactions on*, 19(12):2012–2021.

TWEETDICT: Identification of Topically Related Twitter Hashtags

Fabian Dreer
CIS, University of Munich
dreer@cip.ifi.lmu.de

Eduard Saller
CIS, University of Munich
sallere@cip.ifi.lmu.de

Patrick Elsässer
CIS, University of Munich
elsaesser@cip.ifi.lmu.de

Desislava Zhekova
CIS, University of Munich
zhekova@cis.uni-muenchen.de

Abstract

This paper presents the TWEETDICT system prototype, which uses co-occurrence and frequency distributions of Twitter hashtags to generate clusters of keywords that could be used for topic summarization/identification. They also contain mentions referring to the same entity, which is a valuable resource for coreference resolution. We provide a web interface to the co-occurrence counts where an interactive search through the dataset collected from Twitter can be started. Additionally, the used data is also made freely available.

1 Introduction

In the last couple of years the use of the metadata tag called *hashtag* has significantly changed the manner of use of contemporary social media. As Tsur and Rappoport (2012) present, a *hashtag* is an unspaced string of characters that is indexed by the hash symbol (#). Hashtags, in the function in which we are here interested in, were first discussed by Messina (2007) in his search for contextualization, content filtering and exploratory serendipity within the social networking and microblogging service Twitter. Only a couple of years after (in 2009), Twitter has initialized the linking of identical hashtags within its microblogs, which was shortly followed by other major social networks and services, such as Facebook, Google+ and Instagram. Hence, hashtags have become a vital part of modern communication, context filtering and organization.

The use of hashtags can often be viewed as being a pointer to a specific topic, indication for the context, or even as a one-word summary of the whole text it occurs in. Recognizing this power and expressiveness of hashtags, social networks targeted the constant monitoring and ranking of often occurring hashtags with the hope to achieve an overview of currently popular discussions and trends in society and even enable the establishment of communities around their distinct interests. Yet, often enough, a number of hashtags are used to refer to different aspects of the same topic and the collection of such can be highly helpful for the purpose of topic identification. Moreover, when labelling a topic, people may select from a range of distinct linguistic expressions to refer to the main topic entity/event/concept/etc. Thus, such collections/clusters of hastags might contain valuable information for coreference resolution.

Hereby, we present TWEETDICT, a system for the automatic identification of topically or entity related Twitter hashtags. The paper is structured in the following way: In section 2, we discuss the use of hashtags for topic representation and coreference resolution. In section 3, we present TWEETDICT and provide details about its architecture, extraction and clustering of the hashtags, after which we provide a discussion (section 4) and then conclude our work in section 5.

https://twitter.com

http://www.facebook.com
https://plus.google.com
http://instagram.com

2 Related Work and Motivation

Twitter hashtags have been employed in a number of NLP tasks so far, mostly related to sentiment analysis, such as (Davidov et al., 2010; Mohammad, 2012; Kunneman et al., 2014). Pöschko (2011) explored hashtags in Twitter microblogs and made use of their co-occurrence, as defined in equation (1), where h_i and h_j are two distinct hashtags and their co-occurrence count C is obtained by observing both hashtags in the same microblog, also called tweet, t.

$$C(h_i, h_j) := |\{t|h_i \in t \wedge h_j \in t\}| \quad (1)$$

Pöschko (2011) uses these co-occurrence counts in order to create a dictionary $D(h)$, where $h = h_i$ and $h \neq h_j$. $D(h)$ is then constructed by the ten hashtags that most often occur with h. The author argues that hashtags, such as #tcot, #p2 and #sgp, consisting only of acronyms or abbreviations or altogether non-standard words are not easily understandable or completely unknown. He points out that one solution for their disambiguation, for example, can be the use of the co-occurrence dictionary $D(h)$, which provides words that are somehow related to h and can serve as a definition for that term. In order to explore the intensity of the relations in $D(h)$ Pöschko (2011) uses WordNet (Miller, 1995; Fellbaum, 1998), but the author himself points out that the lexical database lacks on coverage since a large number of hashtags are rather tokens that are not contained by the lexical database.

Our hypothesis, however, is that searching for the intensity or exact type of semantic relation between any number of hashtags is not going to be very indicative of their actual semantics, because of the simple manner of use of hashtags, which as we pointed out in section 1 is often a keyword of a specific topic or a one-word summary of the whole text it occurs in. Following, often co-occurring tags are semantically not related, in the classical understanding of semantic relation (e.g. hyponymy, meronymy, antonymy, synonymy, etc.), but rather bound by the fact that they are both keywords for an existing topic. Based on this hypothesis, we argue that clusters of co-occurring hashtags can be highly helpful, yet, these clusters will serve not as a definition of unknown hashtags, but rather as identificators for the topics this hashtag occurs in.

Topic detection or representation is, yet, not the only area such clusters can be used for. Coreference Resolution (CR) is also a NLP application that is currently heavily demanding flexible, wide-coverage and easily available world knowledge. Ontological information is generally used to represent such knowledge, but when it is manually collected it does not reach the needed coverage for the CR task or in case of an automatic ontology creation it is either not precise enough or collected from resources that do not necessarily contain most recently introduces concepts and entities. A good example, is again WordNet, which contains entities, such as *Barack Hussein Obama* as an instance of *President of the United States* or *Anthony Hopkins* as an instance of *actor*, but *Jack Nicholson* as many other proper names are not covered by the largest ontology for English.

Another automatically created resource for such knowledge is the recently released Wikipedia Links Corpus (Singh et al., 2011), a collection of 43 928 entities (1 567 028 mentions), yet, during the corpus creation mentions with large string edit distance (e.g. President – Barack Obama) were completely discarded in order to avoid noise in the data. As discussed in (Zhekova et al., 2014), this leads to a collection of trivial pairs with large string overlaps (e.g. President Obama – Barack Obama). However, most state-of-the-art CR systems monitor exactly string overlap between the mentions during resolution and thus for them such pairs are not very helpful. We assume that for a given search term h, co-occurring hashtags have a high chance of containing mentions that refer to the same entity, but have low or none string overlap with the target mention (e.g. President – Obama). Extracting such pairs from Twitter is an invaluable resource for CR, because Twitter's microblogs contain discussions about the newest topics and respectively often provide the first mentions of new entities.

3 TWEETDICT

The TWEETDICT system is a Python implementation that, following Pöschko (2011), given

`http://wordnet.princeton.edu`

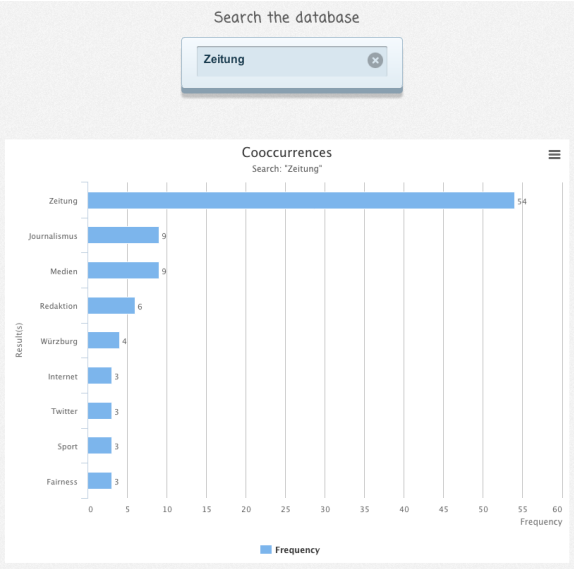


Figure 1: TWEETDICT's web interface.

a search term (a target hashtag) explores microblogs and extracts hashtags that co-occur with that search term in them. In general, the implementation can be applied to any language for which tweets containing hashtags are currently accessible, however, during development and testing we restricted TWEETDICT's functionality to a particular dataset (see section 3.1).

3.1 Data and Accessibility

TWEETDICT makes use of the freely available Twitter REST and Streaming APIs, which are employed for the extraction of the tweets. In order to restrict the dataset to a manageable amount of data we only collected microblogs from a particular target group – followers of the German news show ZDFheute (@ZDFheute) – based on the assumption that these will be interested in and discussing mainly current topics that have been introduced in the show. Thus, the current collection of hashtags does not cover all hashtags in use. There is no further language restriction integrated in TWEETDICT. In fact, the system can be used with an arbitrary collection of tweets and the larger this collection is, the more representative the resulting clusters are.

Altogether the collected data sums up to a set of 7.2 GB for 326 750 hashtagged microblogs (tweets that contained less than 2 hashtags were not considered at all) produced by 34 054 users. The tweets were collected between April 13 and April 19, 2014 as all tweets produced by a follower were extracted.

3.2 Hashtag Extraction and Preprocessing

In order to provide an efficient interface and search capabilities for the system, the co-occurrence counts needed to be preprocessed and stored in a static form. The latter consists of the pairs of co-occurring hashtags plus additional information about the microblogs kept along, e.g. the tweet ID in which the pair occurred. A web interface to the co-occurrence counts is already available (shown in figure 1) and we also release the preprocessed dataset (reduced to the size of 30 MB), available from TWEETDICT's website.

Yet, the interactive search on TWEETDICT's web interface only displays one single cluster containing all hashtags co-occurring with the target one ranked based on their frequency of occurrence. For topic representation and coreference resolution, however, such a cluster is not very helpful. All co-occurring hashtags often represent a wide range of topics or references to a number of distinct entities. Thus, an extended model was generated that aims to provide better expressiveness for these tasks (described in section 3.3).

3.3 Clustering

In order to tackle the expressiveness problem (see section 3.2), which goes beyond Pöschko's proposed dictionary representations, we extend the system with a recursive search through all hashtags in the initially generated cluster. This means that the system initializes a search based on a given search term and then uses the resulting dictionary as seeds for consequent searches. In this manner the data can be exhaustively explored and a graph consisting of multiple interconnected clusters can be built based on all hashtags occurring in the tweets. An example graph is displayed in figure 2. For the visualization of the graph, the software version control visualization

https://dev.twitter.com/docs/api
https://dev.twitter.com/docs/api/streaming

http://tweetdict.cis.uni-muenchen.de
http://tweetdict.cis.uni-muenchen.de/hashtags.json

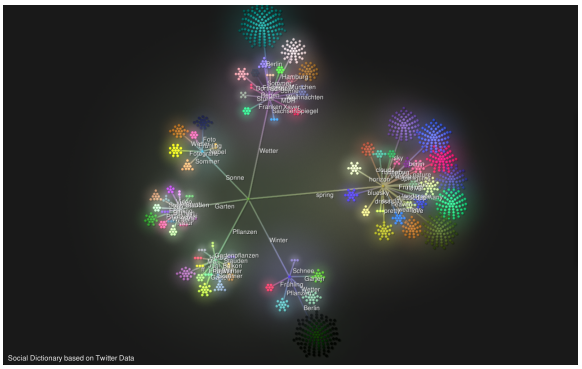


Figure 2: An initial stage of a graph created via a recursive search through the data.

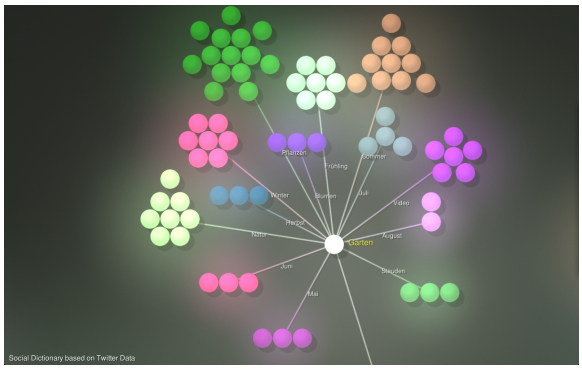


Figure 3: Zoomed-in part of the graph.

h	$D(h)$
Ukraine	Krim, Russland, Putin, Russia, Crimea
NSA	Snowden, Obama, Merkel, Überwachung, Heartbleed
android	androidgames, gameinsight, flappybirds, mariobross, app
Zeitung	Journalismus, Medien, Redaktion, Wrzburg, Internet

Table 1: Example clusters ($D(h)$) per target hashtag (h). For simplicity, the # symbol is left out.

tool Grouce was made use of.

For the purpose of cluster generation, only hashtags that co-occur more than 10 times with the target are included and the graph is restricted to extensions of at most two levels of subtrees per given search term. In order to allow the separation of topics, namely, that one search term can be used for a number of topics, its occurrence across the formed clusters is not restricted. Yet, to avoid infinite loops in the recursion, self-references and back-references are not followed further.

4 Discussion

As can be well seen on the zoomed-in image of the graph provided in figure 3, the resulting clusters may consist of a considerably different number of nodes. According to our preliminary qualitative observations, larger clusters tend to still contain a mixture of topics, while smaller clusters consist mainly of coreferential or highly related tokens (tokens refering to one topic).

We assume that such large clusters can be subdivided based on significance tests between the difference of frequency distributions across the cluster. Hashtags referring to the same topic or entity will potentially be used a similar number of times.

The results returned by TWEETDICT visualized in table 1, show that co-occurring tags may also be in languages other than the target language, e.g. the pair *Ukraine* (German) – *Russia* (English). This is a result of the fact that hashtag use is not restricted in any way apart from the general syntactic constraints, which allows users to combine hashtag translations when they post a microblog containing both languages.

5 Conclusion and Future Work

In the current paper, we presented TWEETDICT, which is a prototype of a system that can be used for the extraction of hashtag clusters based on co-occurrence of hashtags in Twitter microblogs. As we noted, these clusters, can be used for a number of NLP applications, such as topic summarization/representation or coreference resolution.

Further on, we plan to explore a number of issues and open questions for the generation and improvement of the clusters and their expressiveness. One such issue is, for example, the targeted filtering of irrelevant or noisy tweets, e.g. tweets that contain more than 4 hashtags or consist solely of hashtags.

Another direction would also be the exploration of hashtags occurring only in tweets of the same language. This will allow for a clearer and language dependent representation.

Additionally, an important issue to look at is the subdivision of clusters based on significant difference of the frequency distributions of the hashtags. This will allow for the generation of even smaller clusters that do not contain a mix-

https://code.google.com/p/gource

ture of topics or entities.

References

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced Sentiment Learning Using Twitter Hashtags and Smileys. In *Coling 2010: Posters*, pages 241–249, Beijing, China, August. Coling 2010 Organizing Committee.

Christiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database*. MIT Press.

Florian Kunneman, Christine Liebrecht, and Antal van den Bosch. 2014. The (Un)Predictability of Emotional Hashtags in Twitter. In *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM)*, pages 26–34, Gothenburg, Sweden, April. Association for Computational Linguistics.

Chris Messina. 2007. Groups for Twitter; or A Proposal for Twitter Tag Channels, in Personal Blog: *FactoryCity*: `http://factoryjoe.com/blog`.

George A. Miller. 1995. WordNet: A Lexical Database for English. *Commun. ACM*, 38(11):39–41, November.

Saif Mohammad. 2012. #Emotional Tweets. In *SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 246–255, Montréal, Canada, 7-8 June. Association for Computational Linguistics.

Jan Pöschko. 2011. Exploring Twitter Hashtags. *CoRR*, abs/1111.6553.

Sameer Singh, Amarnag Subramanya, Fernando Pereira, and Andrew McCallum. 2011. Large-scale cross-document coreference using distributed inference and hierarchical models. In *Proceedings of the 49th Annual Meeting of the ACL: Human Language Technologies*, pages 793–803, Portland, Oregon, USA, June. ACL.

Oren Tsur and Ari Rappoport. 2012. What's in a Hashtag?: Content Based Prediction of the Spread of Ideas in Microblogging Communities. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, WSDM '12, pages 643–652, New York, NY, USA. ACM.

Desislava Zhekova, Robert Zangenfeind, Alena Mikhaylova, and Tetiana Nikolaienko. 2014. Alignment of Multiple Translations for Linguistic Analysis. In *Proceedings of the The 3rd Annual International Conference on Language, Literature and Linguistics (L3)*, Bangkok, Thailand, 9 - 10 Juni. (to appear).

Sentilyzer – A Mashup Application for the Sentiment Analysis of Facebook Pages

Hartmut Glücker, Manuel Burghardt, and Christian Wolff
Media Informatics Group
Institute for Studies in Information and Media, Language and Culture
University of Regensburg
firstname.lastname@ur.de

Abstract

We present *Sentilyzer*, a web-based tool that can be used to analyze and visualize the sentiment of German user comments on *Facebook* pages. The tool collects comments via the *Facebook API* and uses the *TreeTagger* to perform basic lemmatization. The lemmatized data is then analyzed with regard to sentiment by using the *Berlin Affective Word List – Reloaded* (BAWL-R), a lexicon that contains emotional valence ratings for more than 2,900 German words. The results are visualized in an interactive web interface that shows sentiment analyses for single posts, but also provides a timeline view to display trends in the sentiment ratings.

1 Introduction

Social media platforms such as *Facebook* or *Twitter* churn out vast amounts of user generated content. This data can be analyzed with regard to subjective information – i.e. people's emotions, attitudes, opinions, and sentiments – to monitor specific topics or detect trends. Such analyses are typically referred to as *sentiment analysis* or *opinion mining* [Liu, 2012].

This article introduces *Sentilyzer*, a web application for the sentiment analysis and visualization of user comments on Facebook pages. The comments are lemmatized and sentiment scores are clustered according to previously defined keywords. The results of the sentiment analysis are presented to the user in an interactive web interface. The rest of the article is structured as follows: Section 2 gives an overview of the technical realization of *Sentilyzer*; section 3 presents the main features and basic functionality of the tool. Section 4 concludes the insights of a first case study that has been conducted with *Sentilyzer*, and also describes the next steps in the development of the prototype.

2 Technical realization of Sentilyzer

Sentilyzer is realized by means of a client-server architecture that requires an *Apache* server with *PHP* and a *MySQL* database. Lemmatization and sentiment analysis are realized on the server-side by using *Java*. *Sentilyzer* can be categorized as a *mashup* application, as it integrates a number of freely available, third-party components in a common web interface:

Data crawler and web interface: *Facebook Graph API* (application programming interface for crawling Facebook data)[1], *Foundation 5.1* (HTML template framework)[2], *Isotope.js 2.0* (JavaScript library for element sorting)[3], *Laravel 4.1* (PHP framework for web applications)[4], *NVD3.js 1.1*

[1] https://developers.facebook.com/docs/graph-api; all URLs mentioned in this paper were last accessed July 10, 2014

[2] http://foundation.zurb.com/

[3] http://isotope.metafizzy.co/

[4] http://laravel.com/

(JavaScript library for facilitated creation of graphs based on the *D3.js* library)[5]

Lemmatizer and POS tagger: *TreeTagger* (POS tagger and lemmatizer for German)[6], *TT4J* (Java wrapper for TreeTagger)[7]

Sentiment lexicon: *Berlin Affective Word List – Reloaded* (BAWL-R)[8]

3 How Sentilyzer works: Basic functionality in five steps

The basic functionality of *Sentilyzer* can be broken down into five basic steps that are explained in more detail in the following sections.

3.1 Preliminaries: Project and database setup (Step 1)

Before *Sentilyzer* can analyze the sentiment of Facebook comments, the user needs to define the basic project details via an *XML* configuration file. First, the name of the *Facebook* page that is to be analyzed needs to be specified. Users can also define a timeframe (start and end date) for posts from this page to be included in the analysis. As *Sentilyzer* allows the user to display aggregated sentiment scores for clusters of comments as well as sentiment trends for such clusters throughout time, it is important to specify the parameters for these clusters in advance. It is possible to define arbitrary *timelines* (=clusters of posts) containing only posts that include or exclude certain keywords:

```
<timeline>
 <name>Michael Wendler</name>
  <includePostsWithKeywords>
   <keyword>Michael</keyword>
   <keyword>Wendler</keyword>
  </includePostsWithKeywords>
  <excludePostsWithKeywords>
   ...
  </excludePostsWithKeywords>
</timeline>
```

[5] http://nvd3.org/

[6] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

[7] https://code.google.com/p/tt4j/

[8] http://www.ewi-psy.fu-berlin.de/einrichtungen/arbeitsbereiche/allgpsy/BAWL-R/index.html

After a new project has been created according to the parameters specified in the XML-configuration file, a corresponding database structure is created automatically by the tool.

3.2 Crawling the Facebook page (Step 2)

In the second step, the crawler component collects all posts and comments from the previously specified *Facebook* page via the *Facebook Graph API*. The following information for posts and associated user comments is stored in the relational database:

Posts: *message text*, *number of likes*, *number of comments*, *number of shares*, *date of publication*

User comments: *author name*, *message text*, *number of likes*, *date of publication*

3.3 Clustering of posts (Step 3)

In this step the tool creates *timeline* clusters of posts according to the keywords that have been specified in Step 1. This clustering of posts allows the user to compare aggregated sentiment scores of different *timelines* (e.g. for different celebrities) in the final step.

3.4 Lemmatization and calculation of sentiment scores (Step 4)

Step 4 contains two important sub-steps: First, the message texts are lemmatized to make them available for automatic sentiment analysis. *Sentilyzer* utilizes an existing lemmatizer for German language, the *TreeTagger* [Schmid, 1994].

Second, the lemmatized comments are compared with a lexicon that contains sentiment scores for different words. For the German language, there are only few resources that can be used as a sentiment lexicon. We identified the *Multi-layered Reference Corpus for German Sentiment Analysis* (MLSA) [Clematide et al., 2012] and the *Berlin Affective Wordlist – Reloaded* (BAWL-R) [Võ et al., 2006, 2009] as appropriate resources for this project. Eventually, we decided to use the BAWL-R lexicon, as it provides more sentiment annotations for single words (over 2,900 words) than MLSA (about 820 words), with the latter being more focused on multi-level sentiment annotation that includes larger units such as *phrases* and *sentences*.

Figure 1: The example shows the original comment and the lemmatized version as well as the BAWL-R sentiment score for a matching word.

The BAWL-R lexicon provides scores for *emotional valence*[9], "ranging from –3 (*very negative*) through 0 (*neutral*) to +3 (*very positive*)" [Võ et al., 2009, p. 535]. The positive and negative values of words that match the BAWL-R lexicon are summed up to an aggregated sentiment score for each comment (cf. Figure 1).

3.5 Visualization of sentiment scores (Step 5)

In the last step, the results are visualized in an interactive web interface. The results are organized according to the *timelines* that were specified in Step 3. All posts of a *timeline* are displayed chronologically and can be sorted with respect to different parameters such as *positive / negative sentiment*, *number of comments*, etc. (cf. Figure 2). Alongside the *message content*, *number of likes*, *number of comments*, *number of shares* and *publication date*, the tool displays the aggregated sentiment score for all comments that are associated with a post. The tool also provides an aggregated sentiment score for all comments associated with a specific timeline as well as a view that shows sentiment trends for comments to different posts in the course of time (cf. Figure 3).

4 Conclusions and outlook

Sentilyzer serves as a proof of concept for a tool that is able to *crawl* user comments from *Facebook* pages, to *analyze* their sentiment, and to *visualize* the results in a user-friendly and interac-

[9]BAWL-R also contains information about *arousal* and *imageability*. This additional information was not utilized in the current prototype, but could be supplemented in a later version of the tool.

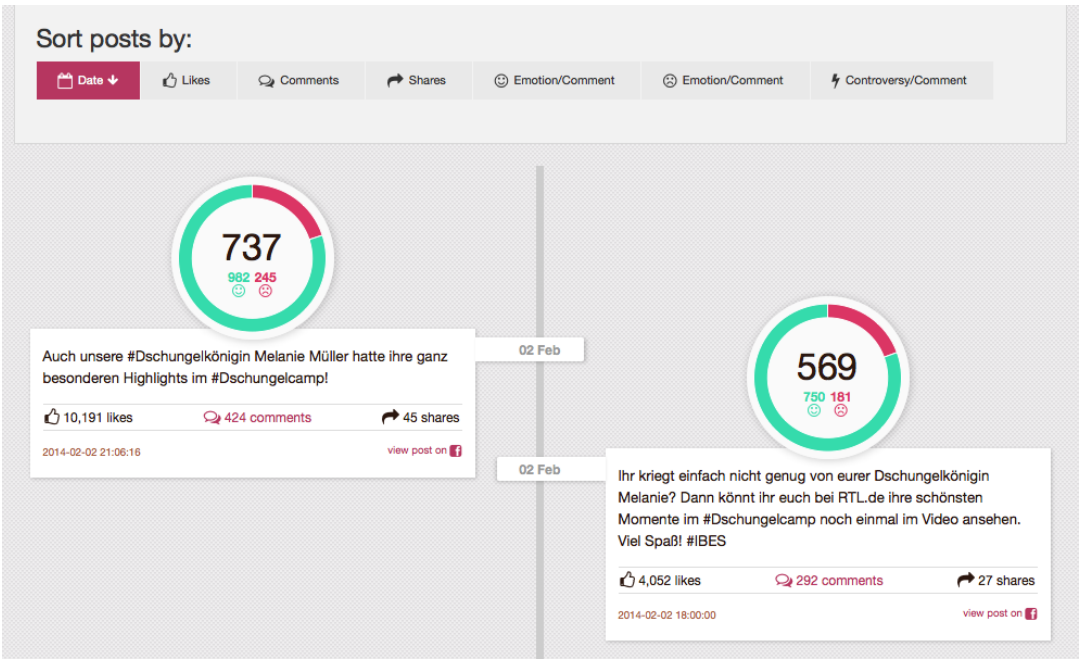


Figure 2: Posts with aggregated sentiment scores for all associated comments. The posts are displayed chronologically be default, but can be sorted by a number of different parameters as well.

tive web interface. As the tool utilizes a number of freely available APIs and tools as well as an existing sentiment lexicon for German, it may be considered a *mashup* application. By using third party components for natural language processing and sentiment analysis of social media data it also becomes obvious that existing resources are not optimized for the specifics of computer-mediated language, e.g. specialized vocabulary and "loose" orthography. We are planning to create a crowd-sourced lexicon with lemmatized forms and sentiment scores for computer-mediated language in an upcoming research seminar on sentiment analysis, thus hopefully improving the current weaknesses of the prototype.

Nevertheless, *Sentilyzer* has already been used successfully to analyze the perception of candidates from the German reality show "Ich bin ein Star - Holt mich hier raus (2014)" on the official Facebook page[10]. The large number of user comments compensated for most of the erroneous lemmatizations and sentiment scores, and could be used successfully to show aggregated sentiment scores and sentiment trends through the course of the TV show.

A live demo of *Sentilyzer* with sentiment visualizations for all candidates is available at `http://dh.wappdesign.net/`. We are currently working on a documented version of the application that will be available via *GitHub* for local installation. In the long-term, we are planning to host *Sentilyzer* as a web service.

[10] `https://www.facebook.com/IchBinEinStar`

References

Simon Clematide, Stefan Gindl, Manfred Klenner, Stefanos Petrakis, Robert Remus, Josef Ruppenhofer, Ulli Waltinger, and Michael Wiegand. MLSA-A Multi-layered Reference Corpus for German Sentiment Analysis. In *Proceedings of LREC '12*, pages 3551–3556, 2012.

Bing Liu. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool, San Rafael, CA, 2012.

Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK, 1994.

Melissa L H Võ, Arthur M Jacobs, and Markus Conrad. Cross-validating the Berlin Affective Word List. *Behavior research methods*, 38(4): 606–609, 2006. ISSN 1554-351X.

Melissa L-H Võ, Markus Conrad, Lars Kuchinke, Karolina Urton, Markus J Hofmann, and Arthur M Jacobs. The Berlin Affective Word List Reloaded (BAWL-R). *Behavior research methods*, 41(2):534–538, 2009. ISSN 1554-351X.

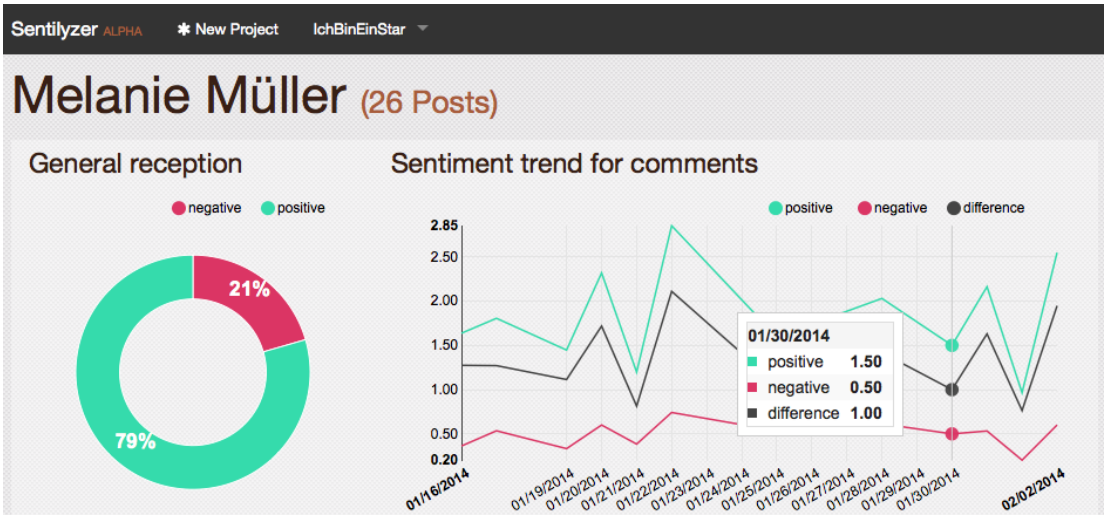


Figure 3: Aggregated sentiment score for all comments associated to a specific timeline and sentiment trend in the course of time.

Alpes4science project : SMS corpus processing and tokenization problems

Eleni Kogkitsidou and **Georges Antoniadis**
Lidilem Laboratory, University of Stendhal,
Grenoble, France

{eleni.kogkitsidou,georges.antoniadis}@u-grenoble3.fr

Abstract

Virtual textual communication involves numeric supports as transporter and mediator. SMS language is part of this type of communication and represents some specific particularities. An SMS text is characterized by an unpredictable use of whitespaces, special characters and a lack of any writing standards, when at the same time stays close to the orality. This paper aims to expose the database of alpes4science project from the collation to the processing of the SMS corpus. Then we present some of the most common SMS tokenization problems and works related to SMS normalization.

1 Introduction

With the appearance of new forms of virtual communication (chats, email, social networks, etc.), new terms have been invented to describe this new type of communication: computer-mediated communication (CMC), written computer-mediated communication or network-mediated communication, cybercommunication, netspeak, etc. Since 90s, SMS communication belongs to this type of communication and it's the subject of our study. The interest to study the SMS communication and the SMS language, in our case, is identified at the particularities which this language presents. It's a discourse that escapes the institutional constraints and lacks any standards (Panckhurst, 2009). As it is mentioned by Barasa and Mous (2009), SMS text is characterized by a rich lexical creativity without conventions, and a creation of a new form of orthography. Stark (2011) described SMS as a strict and particular writing code which combines several methods to shorten sentences and words. On the other side, it is close to the orality by remaining a written form and that's why this kind of language is a subject of interest for many researchers (Antoniadis et al., 2011).

2 The alpes4science project

The observation of these particularities requires authentic and certified materials in order to obtain an objective view (Fairon and Paumier, 2006). The sms4science[1] project aims to respond to this need by launching, in 2004, the first collation of SMS at CENTAL[2] laboratory of Catholic University of Louvain, and establishing a collation methodology and protocols for SMS corpora construction. Since then, several other works related to this project have been released (Reunion Island, 2008, http://www.lareunion4science.org/; Switzerland, 2009, http://www.sms4science.ch/; Quebec, 2010, http://www.texto4science.ca/; Montpellier, 2010, http://www.sud4science.org/) (Panckhurst, 2013).

Our study uses as starting point the SMS corpus of alpes4science[3] project which is the part of sms4science project. The alpes4science project was signed in 2009 between LIDILEM[4] and the

[1] http://www.sms4science.org/
[2] Center of Natural Language Processing
[3] www.alpes4science.org
[4] lidilem.u-grenoble3.fr/

General Council of Hautes-Alpes for the purpose to create a database.

The collation took place from 1 October 2010 to 31 January 2011 in Hautes-Alpes and Isère of France. For this reason, the topic of messages is related to local and seasonal events (snow, ski, pistes, end of year celebrations, greetings etc.). However, we identify some sent messages which were saved in the mobile phone and they are not related to the chronological period of the collation, such as for example messages like : "thanks", "see you later" (Chabert et al., 2012).

In total, 359 people sent their 22054 SMS to the platform. Each participant should send his messages to a special number by writing the "SMS05" code at the beginning of every sent message. Thereafter, all messages were transported to a special dedicated platform. The registration was done once the participant had sent his first message beginning with the "SMS05" code and following his phone number. In this way, participants were automatically associated with an identification number and they could transfer their messages (Antoniadis et al., 2011).

The participants of the project were invited to complete a questionnaire with varied information concerning their social profile (age, gender, education level, profession, mother tongue etc.), as well as, their communicative character (texting frequency, keyboard, language register etc.). Among participants 119 persons didn't answer the questionnaire. As for the rest of 240 persons we know that the 70.8% represents female SMS writers and the 29.2% male writers aged from 14 to 69 years old. This metadata is an incontestable material for the production of scientific studies through the analysis of this information in the fields of linguistics, natural language processing, sociology and sociolinguistics for the purpose of establishing actual observations.

2.1 Corpus Processing

With the construction of the SMS corpus we can examine adequately the function of languages and explore exhaustively authentic language productions. In our case, we focus on the original SMS corpus which allows us to examine the particularities of this type of communication. There are two types of treatment that are essential to make the SMS corpus operational and able to give way to other NLP applications (Sproat et al., 2001; Beaufort et al., 2010) : the *anonymization* of sensitive data for ethical reasons and the *transcription* that aims to make readable and usable messages in order to facilitate the operation of the corpus.

2.1.1 Corpus anonymization

The anonymization of data doesn't exclusively concern SMS messages but also any other form of communication and data type (state protected data, University restricted or critical data, telecommunications, electronic commerce, etc.). This is a compulsory process by ethics and by agreement with the CNIL[5] (1442138) for the authorized diffusion of corpora in order to preserve the confidentiality of transmitted information. In alpes4science corpus we consider as sensitive data: last names, nicknames, surnames, phone numbers, e-mail addresses, URL, codes, postal addresses, as well as, any other information which allows the indirect identification a person. The anonymization process had been achieved via a web interface designed for this project which was capable to detect standard format data (for example: e-mail addresses, URL, phone numbers), then, three researcherswere in charge to verify the result which were automatically produced. The data to be anonymized was replaced by a new form. This new form matched ***(DATA NAME)_Number of data character*** (table 1).

Original SMS	j'écris à Mathieu
Anonymized	j'écris à ***SURNOM_7***
Translation	I'm writing to Mathieu

Table 1: Anonymization example

2.1.2 Corpus transcription

The transcription of SMS aims to make a message which contains abbreviations, phonetizations, extensions etc. understandable to everyone. Before proceed to the SMS transcription we had defined, in a strictly way, through a protocol all the elements which meant to be modified from the original message to the standard language. The

[5]http://www.cnil.fr/english/

purpose of this processing is to release a minimum of changes and only if it is necessary (table 2).

Original SMS	Oui bien sur qan tu veu
Transcription	Oui bien sûr quand tu veux

Table 2: transcription example

The applied methodology consists of trancoding manually SMS which from their part contribute to create a dictionary to the database with SMS words. This method proposes subsequently to the researcher the possibility to make a choice to keep or change the word to by transcription via a web interface.

3 SMS tokenization problems

Tokenization process for "standard" alphabetic languages is defined as the division of character sequences into sentences and sentences into tokens. As tokens we consider words, numbers and every other punctuation marker. Although, Dale (2000) gives us a simple definition of text tokenization process without taking into account punctuation markers or numbers: *Tokenization is the process of breaking up the sequence of characters in a text by locating the word boundaries, the points where one word ends and another begins*.

The importance of this process for Natural Language Processing (NLP) applications such as POS taggers, parsers, search engines, text normalization etc. is because they deal with words and sentences. Most tokenizer applications use a simple method which implements words separations by blanks, thus a white space is a delimiter of word boundaries and also separate punctuation markers (Schmid, 2007). For alphabetic languages the main problem of tokenization is the ambiguity between abbreviation periods, multiword expressions, sentence markers, etc. (fig., etc., U.K., S. Africa, have fun).

It is already hard to delimit the boundaries of a "standard" alphabetic language token, with regard to SMS language we release that segmentation of tokens becomes a real "challenge". To these standard tokenization problems joins SMS tokenization problems with graphical, phonetical and morphological particularities. An SMS text is characterized by an unpredictable use of whitespaces, special characters and a lack of any writing standards. SMS word is not always surrounded by whitespaces, punctuation marks are usually absent and special marks, such as emoticons, are frequently used.

We summarize below some SMS problems which need to be solved :

- Multiword non-standard abbreviations: tokens which borrow the initials of a multiword expression ex. lol = laugh out loud, stp = s'il te plait (please)
- Sentence boundary detection: most of the time a punctuation mark is missing at the end of a SMS sentences
- Missing whitespaces and punctuation marks: abbreviations promote the omission of an apostrophe or a whitespace between two or three words which generate semantic ambiguities ex. ct= cette (this), ct= c'est (it is)
- Other punctuations – Emoticons: it's about symbolic figures composed by punctuation marks and letters which represent a graphical form of emotions ex. :) = smile, ;) = winking
- Mix of characters and numbers: SMS words are usually composed by numbers and characters ex. 2day= today, dem1= demain (tomorrow)
- Extending punctuation marks: commonly used in order to express a large wonder, admiration, the thought or happiness and sadness with emoticons ex. quoi??????? (what???????), :)))))))))

3.1 From tokenization approaches to SMS normalization

The fundamental step of a text pre-processing is the normalization of a text. Sproat el al.(2001) insist in the fact that normalization must be applied before any other classic NLP process. Most of the time, normalization involves tokenization process. As it concerns SMS, text tokenization is a trivial processing stage. Normalization process of SMS aims to convert informal text in a grammatically correct text. Non standardized SMS

message is represented as a sequence $T = T_1, T_2, ..., T_n$ of tokens. As a given token T_i, we define the operation of normalization R, such as $R(T) = r_1, r_2, ..., r_n$ is a set of normalizations of T:

Given T_i= *combien (how many)*
R (combien) = cmbien, cb, cmb, kmbien, cbien

There are three approaches till now in order to achieve an SMS normalisation : a) spell checking, b) machine translation and c) automatic speech recognition (Kobus et al., 2008). Beaufort et al. (2010) propose a hybrid rule which combines both of these approaches spell checking and machine translation. These methods are based on models learned from a SMS aligned at character level corpus and its transcription. With the purpose of tokenizing Twitter messages which are similar to SMS messages, Kaufmann and Kalita (2010) use a two step model that fist preprocess messages to remove noise and they feed them into a machine translation model in order to convert them into standard English. Although, neither Kobus et al.(2008) nor Kaufmann and Kalita (2010) take into account phonetic similarities which are frequently presented. Han et al. (2011), at the other side, use a cascaded method which detects bad-formed words and generates candidates based on morphophonemic similarities. An alternative approach offers Aw et al. (2006), by a different point of view, he consider normalization as a translation problem and adopt a method which aims to adapt a phrase based statistical machine translation model. Choudhury et al. (2007) propose the application of a model in which the system of normalization uses statistical methods spelling correction conversion based on HMM (Hidden Markov Models) between texting and the standard language. This model was used to construct a decoder SMS text in English to their standard English forms with an accuracy of 89% at the word level. On the same model is based Lopez et al. (2014) in order to obtain a semi-automatic alignment method messages in order to build a dictionary SMS.

Most of the applied studies are based on deterministic techniques for automatic construction of transcription dictionaries, statistical methods for the automatic transcription of a SMS word and analysis of hybrid approaches (deterministic-probabilistic). Our aim is to focus on transcription process from SMS messages to standard french language. As starting point, of our research we consider that every SMS word refers to a standard language word and there is always a standard word definition for SMS words. We examine multiple different graphical forms of a SMS word by giving the definition of the term *polygraphy* which means that a SMS word can be transcribed in two or more standard words. At the same time, a standard french word can be transcribed in two or more SMS words. Of course, we couldn't omit the fact of the correspondence of one SMS word to one standard word. To this day, these graphical aspects are poorly developed in the SMS related literature (Fairon and Paumier, 2006; Beaufort et al., 2010; Cougnon and François, 2011; Panckhurst, 2009). These observations permit us to have a global view of the ambiguity level that we face in SMS transcription. The goal of our study is to achieve a transcription approach of SMS words to standard language word by applying a rule-based model.

4 Conclusion

In this paper we have presented the alpes4science project from the collection to the processing of SMS messages. Based on SMS language particularities we had defined the tokenization problems and penetrate into normalizations approaches. The alpes4science database is a composition of 22,054 authentic text messages which had been semi-automatically proceed. As a result we dispose an aligned corpus of SMS messages with their transcription, anonymization and segmentation, a dictionary with the couple of SMS words and translation and metadata of the participants' social profile. This material composes an indisputable tool for sociolinguistic and linguistic researches, as well as for NLP applications (automatic name entity extraction, normalization, information retrieval etc.). The processing of the SMS corpus allows us this day to expect the upcoming online publication of the corpus by the Consortium of written corpus, of CoMeRe project.

5 Acknowledgment

Funding for this project was provided by a grant from *la Région Rhône-Alpes*.

References

Antoniadis G., Chabert G., and Zampa V. 2011. Alpes4science: Constitution d'un corpus de SMS réels en France métropolitaine, talk. *79th Acfas colloquium*, Sherbrooke, May 9-10, 2011.

Aw A., Zhang M., Xiao J. and Su J. 2006. A phrase-based statistical model for SMS text normalization. In *Proc. COLING/ACL 2006*.

Barasa S. and Mous M. 2009. The Oral & Written Interface in SMS: Technologically Mediated Communication in Kenya. *Low Educated Second Language and Literacy Acquisition*.

Beaufort R., Roekhaut S., Cougnon L.-A., Fairon C. 2010. Une approche hybride traduction/correction pour la normalisation des SMS Richard. In *TALN 2010*.

Chabert G., Zampa V., Antoniadis G., Mallen, M. 2012. *Des SMS Alpins* Editions de la Bibliotheque départementale de Hautes-Alpes.

Choudhury M., Saraf R., Jain V., Mukherjee A., Sarkar1 S., and Anupam Basu. 2007. Investigation and modeling of the structure of texting language. *International Journal on Document Analysis and Recognition*, 10(3):157– 174.

Cougnon, L.-A., and François, T. 2011. Etudier l'écrit SMS - Un objectif du projet sms4science. In *Linguistik online 48*.

Dale, R. 2000. *Handbook of Natural Language Processing* (p. 964).

Fairon C. and Paumier S. 2006. *Le langage SMS* . Louvain-la-Neuve: Presses universitaires de Louvain.

Han, B., and Baldwin, T. 2011. Lexical Normalisation of Short Text Messages : Makn Sens a # twitter. In *HLT '11 Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* .

Kaufmann J. and Kalita J. 2010. Syntactic normalization of Twitter messages. In *International Conference on Natural Language Processing*, Kharagpur, India.

Kobus, C., Marzin, P., and Lannion, F. 2008. Normalizing SMS : are two metaphors better than one ? *In: COLING 2008*.

Lopez C., Bestandji R., Roche M. and Panckhurst R. 2014. Towards Electronic SMS Dictionary Construction: An Alignment-based Approach *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*.

Panckhurst R. 2013. A Large SMS Corpus in French: From Design and Collation to Anonymisation, Transcoding and Analysis. In *5th International Conference on Corpus Linguistics (CILC2013)*.

Panckhurst R. 2009. Short Message Service (SMS) : typologie et problématiques futures, in *Arnavielle T. (coord.), 33–52*.

Sproat R., Black A.W., Chen S., Kumar S. , Ostendorf M. , and Richards C. 2001. Normalization of non-standard words. *Computer Speech & Language*, 15(3):287–333.

Stark E. 2011. La morphosyntaxe dans les SMS suisses francophones: Le marquage de l'accord sujet – verbe conjugué *Linguistik Online, 48(4):35-47*.

Schmid, H. 2007. Tokenizing. *Anke Lüdeling and Merja Kytö* (Corpus Lin., pp. 1–17). Mouton de Gruyter, Berlin.

ISCALPEL

Bottom up specialized phraseology in CLIL teaching classes

Elisa Corino
Università di Torino
Dipartimento di Lingue e letterature straniere e Culture moderne
Via San'Ottavio 20
10124 Torino
elisa.corino@unito.it

Abstract

When dealing with language for specific purposes (LSP), teachers always have to confront with issues which are strictly linked to the specificities of the language of a given field. This is particularly true for CLIL teachers in Italy, who are subject teachers sharing with language teachers some aspects of pupils' language education; though, not being prepared to lead students through a path of language awareness and analysis.

This is why these people should be trained in analyzing the features of language and recognizing recurrent lexical and syntactical paths which distinguish specific textual genres or discourse, in order to let their students develop autonomous language capabilities in turn.

Familiarizing with corpus-based procedures turns out to be one of the most useful tools at these teachers' disposal to enquire LSP peculiarities and to find out patterns of specialized phraseology, which are barely mentioned in the general bilingual and monolingual dictionaries used by their students.

Corpus-based methodology in CLIL classes means to empower both teachers and students to develop competences in moving away from mere surface features of text to selecting and understanding meanings and structures, thus using texts with specific intentions and becoming familiar with lexicographic tools such as corpora to compensate the defects of general dictionaries.

1 Introduction

One of the basic principles of Content and Language Integrated Learning (CLIL) is to implement language-aware instruction, which should naturally lead to content-aware instruction. As Ting (2011) reported, that focus on language positively supports content comprehension has been pointed out even by science educators recognizing that language is the access key to content. In particular Snow (2010) acknowledges the language of science to be 'alienating', if not downright annoying, and in fact when teachers adopt that concise and authoritative tone to explain strange-sounding phenomena which young minds could neither see nor fathom, they might transform even the mother tongue into a foreign language. The context thickens when dealing with 'alienating' language for specific purposes (LSP) in a foreign language where the development of a language-aware content education is strictly required.

It is thus clear that content teachers, right before their pupils, should be trained in developing defined competences as well as a general capacity to deal with linguistic settings and requirements that are not fully predictable. (Richards and Farrell, 2005; Tsui, 2003). On this point Hütter et al. (2009) quote teacher education as an "interface of theory and practice", suggesting to train future teachers to work with and analyze LSP texts within an applied linguistics framework in order to prepare them to mediate these insights to language and teaching practice.

Dealing with CLIL implies a deep knowledge of lexico-grammar elements associated with the different domains and disciplines, everyday language can assume different and extremely precise meaning when contextualized in a LSP environment. In economics texts, for example, we find words like *isocost*, *utility*, and *duopoly* occurring frequently; they are unlikely to occur at all or with high frequency in other kinds of texts with

the same meaning. One has to know syntagmatic relationship between words, semantic associations (collocations and prosodies), lexical bundles, besides a specific textual organization (Durrant 2009, Nelson 2006, Gledhill 2000).

In fact, competence in LSP means to master different aspects - lexico-grammatical features, patterns of textualisation, and genre-structuring features or 'moves' - which are relevant to the foreign language learner who needs considerable information regarding the appropriateness and acceptability of particular linguistic choices in individual genres. And some pieces of information are not to be found either in paper or in e-dictionaries (cf. 3), or in even in translation tool kits (i.e. Google translator tool kit, which is extremely popular among students), whereas more detailed information on lexico-grammatical features - such as syntactical markedness and nuances in meaning of near-synonyms - is possible through the use of corpus linguistics, another area of linguistics whose undoubted importance has been reflected also in language teaching, as pointed out by McEnry and Xiao (2011).

A corpus-based bottom-up approach can foster LPS competence of both content teachers and students, by offering facts of actual language usage which are hard to come by with other means (Mindt 1997, Gavioli 2005, Hütter et al. 2009, Walker 2011), especially with regard to typical choice of words (sorting them by frequency), meaning nuances and appropriate use of collocations.

Following this methodology, subjects involved in CLIL education familiarize with the potential of specialized corpora, learning how to use them as a tool in materials development and as special lexicographic source which is tailored to their LSP needs. It is a way of introducing a kind of Computer-Aided/Assisted Language Learning (CALL) in subjects where it has not been considered yet, using computational methods and techniques not only for language learning and teaching but also to pass on subject contents.

2 CLIL classes and LSP

As pinpointed by Coonan (2007) "the difficulties related to the discipline concern the conceptual complexity of the subject which is compounded by the fact that input and tasks are mediated through the L2".

Learners face a considerable effort for learning new meanings, new textual organization, understanding processes, making distinctions and often deducing information not explicitly stated; on their side content teachers don't know how to affectively select the language peculiarities they have to present to scaffold their students.

CLIL comprises many different disciplines, ranging from neuroscience to history, which means for each subject teacher the necessity to be well-aware of the differences between LSP and the common use of language, as for word frequency, nuances in meaning, syntactic preferences and textual organization. Scientific and academic texts represent a different genre compared to contracts of sale, business applications or literary passages and focus on the language is necessary so that the student can acquire and manifest competence on the content and recognize and use terms and structures specific of each field.

The most frequently mentioned aspect concerns lexis, specifically the lexis of the discipline that has repercussions on the syntactical patterns and obviously on the learning of the content itself. Even though there is evidence of a strong relationship between vocabulary knowledge and reading comprehension ability (Coady 1993), research (Barnett 1986) long ago demonstrated that vocabulary is only one of the variables involved in language competence, and that knowledge of syntax and textual cohesive devices are also related to successful comprehension as defined by recall. What is therefore necessary when dealing with CLIL and LSP is processing all those relationships at the sentence level and intersentential level in order to connect pieces of information or meanings of words and thus synthesize the overall meaning (Chun and Plass 1996).

Teachers are often limited when it comes to effectively introducing and rehearsing new language. Furthermore, strategic, cognitive

language training is something most subject teachers either don't know how to teach or don't have time for in class, so they rely on bilingual word lists and vocabulary matching exercises which seem an attractive shortcut because it takes less time than contextual presentation and yields excellent short term results, whereas long term retention is often disappointing (Walker 2011). A preliminary systematic analysis of the most important aspects of the L2 word learning problem, that is to say, selecting the relevant vocabulary (which and how many words) and creating optimal conditions for the acquisition process is therefore highly desirable.

2.1 Differences in collocational behaviour

As Firth (1968:179) pointed out, "you shall know a word by the company it keeps".

Gaskell and Cobb (2004) stress the importance of working on concordances to reveal grammatical patterns besides vocabulary objectives to define the syllabus. This is particularly important for CLIL lessons because each textual genre and subject is marked by its own 'collocationality' index (Kilgarriff 2006). Words of specialized fields have a particularly strong tendency to occur in collocations, or are most 'collocational', even though their collocates might not be shown in dictionaries.

A bottom-up approach which is aimed at discovering the collocational behaviour of key lexis can be used to answer many other questions. Such an approach can reveal the different senses of a word and show how it may be associated with a particular semantic prosody (as defined in Louw 1993). By studying the collocations associated with a group of so-called synonyms it is often possible to identify slight but significant differences in the meaning of the words in the group, thus fostering language awareness (Gavioli 2005) and noticing processes (Schmidt 1990). Furthermore students are exposed to redundant information and multiple examples of foreign language structures which help them understand how to use constructions they might have had troubles with at first, as proved by Gaskel and Cobb's (2004) work.

Nonetheless it is a process that should be set out by the teacher himself first for two main reasons:

i. language training for himself and consciousness of the possible difficulties students could encounter

ii. selection of the language objectives and contents that should be presented

In fact, while concordances for lexical and even collocational information are quite easy for learners to interpret and for instructors to set up, grammatical concordances may be less so. A grammar pattern is normally distributed, and grammatical patterning may be fairly tricky for learners to extract from a corpus or even to interpret when extracted for them (Vannestal and Lindquist 2007).

Some studies such as the one reported in Walker (2011) prove, for instance, how a corpus-driven approach can help in choosing between semantically-related verbs (e.g. *head, run, manage*) and nouns (e.g. *system, process, procedure*) taken from a LSP domain - namely business English, giving evidence of their collocational behaviour, thus enabling teachers to suggest students the best item fitting different contexts. In a corpus analysis carried out on the BNC it turned out that there are differences in meaning which reflect different styles and convey different approaches in management: based on corpus evidence both the word *run* and the phrase *in charge of* seem to be associated with power (e.g., *run the show, in charge of the country*) and therefore a top-down management style. In addition, the data show that *run* frequently occurs with nouns which describe non-human entities and may give the feeling to the native-speaker audience that their new masters regard them as automatons who simply have to be told what to do. On the contrary the verb *manage* or a phrase such as *responsible for* do not seem to carry the same connotation of power and are more frequently associated with people.

This example perfectly fits the possible contents of a CLIL unit in Economics and clearly demonstrates that many collocations are not simply arbitrary or idiomatic combinations of words. Especially in CLIL contexts teachers should master the tools that might help to disambiguate the different uses of a word and identify slight but significant differences in meaning between what might appear to be groups of synonyms, but differentiate in their prosody and connotational association;

information that is often neglected in dictionaries, Computer Assisted Language Learning (CALL) tools and translation kits.

3 CALL and dictionaries

Intelligent Computer Assisted Language Learning (ICALL) systems inherently provide more learner control than traditional CALL programs due to their sophisticated answer processing mechanisms and are theoretically more CLIL-oriented and suitable than traditional CALL. Unlike the more conventional drill and practice programs, ICALL software employs Natural Language Processing (NLP) which overcomes the rigidity of the response requirements of traditional CALL (Heift, 2002) thus scaffolding language comprehension and learning through interaction with the learner. Furthermore, ICALL should have the potential "to raise awareness of the variety of strategies available and to allow students to make informed choices about the approaches most useful to them" (Bull 1997, cited in Arispe 2014), just as a corpus-based approach would.

It is true that electronic dictionaries and ICALL tools are currently in the process of merging into full-scale lexicographic information tools offering more than just word-to-word translations or paraphrases for a given lemma. Nonetheless users are asked to formulate their own hypotheses and make decisions among a range of possible options given by the tools. Few of them offer support for the choice, LangBot (Arispe 2014) for example gives some words in context to help users choose, but it rather acts as any online translator and is not suited to deal with any phraseological pattern, idiomatic phrases or colloquial expressions; it is best used at the simple word level or when one wants the meaning of a complex - though unmarked -sentence.

Reporting their experiences with EFL learners using dictionaries to decode foreign language texts, both Augustyn (2013), Marello (2014) and Corino (forthcoming) notice that most of them entirely rely on translation, as they choose to type literally on their electronic devices (whether apps or online dictionaries) every utterance they do not understand in L2, or want to produce in the L2, as if they were using a translation tool such as Google Translate, which highlights a lack of proficiency and severe difficulties learners in looking up words in dictionaries.

What is important for CLIL purposes is the lack of NLP tools which take into consideration the different specialized languages with their shades of meaning and connotative implications, with respect to students' habits to widely rely on these language mediators.

If language teachers are getting used to integrate tools that provide scaffolding tutorials and language practice in and out of the classroom, disciplinary teachers are still to be trained as for (I)CALL; the result is that to understand LSP language students often turn to popular tools of machine translation which - though improved - provide pseudotranslation without analysis of grammar or meaning with an "output inevitably peppered with howlers" (Pullum 2013) students seem not to be sensitive to.

Let us consider the field of physics and Italian word *velocità*, for instance, that has two different translations in English: *speed* and *velocity*, meaning two different content concepts.

If we compare the parallel texts produced by Google Translator the problem becomes immediately clear: in the first question *velocità scalare* and *velocità vettoriale* are translated *speed* and *velocity* respectively, but in the following line both of them are referred to as *velocity*. So which should be here the right word?

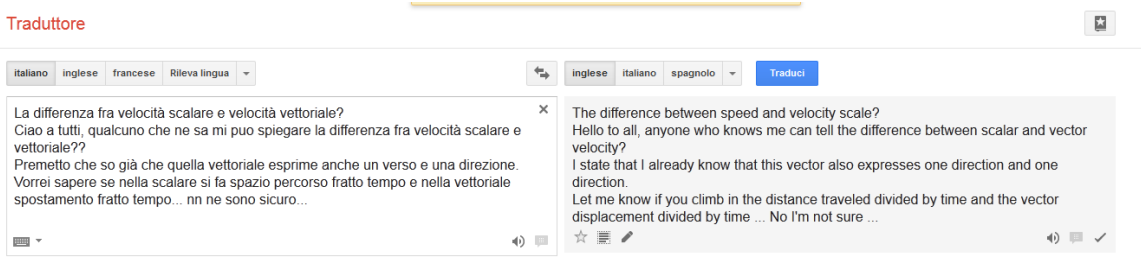


Figure 1. Speed and Velocity according to Google Translate

Of course the translator offers the possibility to substitute the word by one of the suggested options, as in a sort of multiple choice exercise (*speed*, *velocity*, *rate*, *pace*, *momentum*), implying the previous knowledge of the semantic content of the word related to the disciplinary content. It could be efficiently used to build up exercises and tests but it is of no use if one has to disambiguate a term, especially if the process should be applied by a student in a complex CLIL context (much worse and almost droll is the translation of the isolated phrase *velocità scalare* > *climb speed*, which totally ignores PoS attribution).

Nonetheless, even the information found in the bilingual dictionary[1] article is not conclusive in order to define the difference between the two items, neither in the Italian>English section nor in the English>Italian part.

♦ velocità
f.
1 (anche fis.) speed; velocity; (velocità di variazione) rate; (ritmo) pace: (fis.) velocità angolare, angular velocity (o speed)

♦ speed /spi:d/
n. UC[U][C]1 velocità; celerità; rapidità; destrezza; sveltezza: the speed of light, la velocità della luce; What was your speed?, che velocità tenevi (in auto, ecc.)?; (autom.) speed limit, limite (massimo) di velocità; (autom.) low speed, marcia bassa; steady speed, velocità costante; at speed, a grande velocità; at full speed, a tutta velocità; maximum speed, velocità massima (consentita); at top speed, a rotta di collo; di gran carriera; di volata; at a breakneck speed, a velocità folle; to reduce speed, ridurre la velocità; to gather (o to pick up) speed, prendere (o acquistare) velocità; wind speed, velocità del vento2 (mecc.) velocità; marcia: Most cars have five forward speeds, per lo più le auto hanno cinque marce avanti; a ten-speed bike, una bicicletta con il cambio a dieci marce3 (fotogr. = shutter speed) velocità dell'otturatore; tempo d'esposizione4 (fotogr.) sensibilità (di una pellicola)5 (slang) droga stimolante (amfetamina, metamfetamina, ecc.)

♦ velocity /vəˈlɒsətɪ/
n. UC[U][C]velocità; rapidità: (mecc.) uniform velocity, velocità uniforme; the velocity of sound, la velocità del suono; (miss.) escape velocity, velocità di fuga; (econ., fin.) velocity of circulation, velocità di circolazione (della moneta)
● (elettron.) velocity filter, filtro di velocità □ (mecc. dei fluidi) velocity head, altezza cinetica □ (econ., fin.) velocity of money = velocity of circulation ➢ sopra (fis.) □ velocity profile, profilo di velocità.

Under the entry *velocità* in Italian both English *speed* and *velocity* are mentioned following the *(fis)* tag, but without examples or other technical references it turns out to be difficult to decide to which context each term refers to. Starting from the English>Italian section does not make the situation less vague as we cannot find any reference to vectors, and the monolingual dictionary (MEDAL) certainly doesn't either, as no LSP use of the two terms are provided for.

4 Corpora for disambiguation in LSP

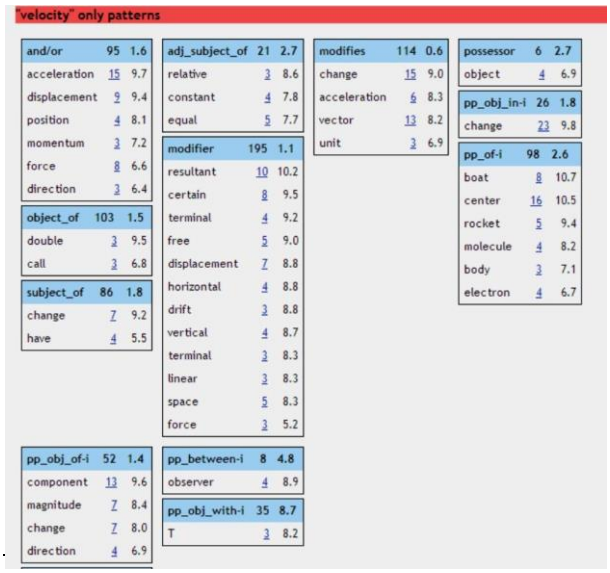
With regard to corpus linguistics, direct use of corpora by learners involves their guided discovery of information about L2 use in corpora (Bernardini, 2004; Leech, 1997). Such

[1] Ragazzini Italian and English dictionary Zanichelli (online edition, www.ubidictionary.zanichelli.it last accessed on 04.09.2014)

an approach can be motivating for learners, and encourages a critical reflection on (prescriptive) grammatical rules or the nuances in meaning of near-synonyms.

One could object that corpora for CLIL purposes should be extremely specific and highly representative, which large generic corpora are not. Tools like the Sketch Engine (www.sketchengine.co.uk) and the web crawler WebBootCat can help in retrieving suitable data and compiling content specific ad hoc corpora.

In the above mentioned case, the disambiguation of *speed* and *velocity* can be solved by compiling a corpus[2] with texts dealing with vector physics and drawing the word sketches of the two words to observe their linguistic behavior. It is then interesting to point out that *velocity* is often modified by *resultant*, *displacement* and *space* (terms generally associated to vector quantity), whereas *speed* is linked through a high frequency number of occurrences to *average* (meaning scalar quantity). *Velocity* followed by the preposition *of* often occur with *center* (talking about velocity of center of mass it is obvious to refer to a vector quantity), while *speed* followed by the same preposition occurs together with *sound* or *wave*, reinforcing the *scalar* suggestion. Comparing the common patterns of the two word sketches it is also to be notice the exclusive occurrence of *speed of light*, conventionally meaning the module of speed, on the other hand *relative* is restricted to the vector quantity.



2 The corpus was created by physics teachers with Sketch Engine and consists of 586,989 tokens.

Figure 2. VELOCITY - Word Sketch

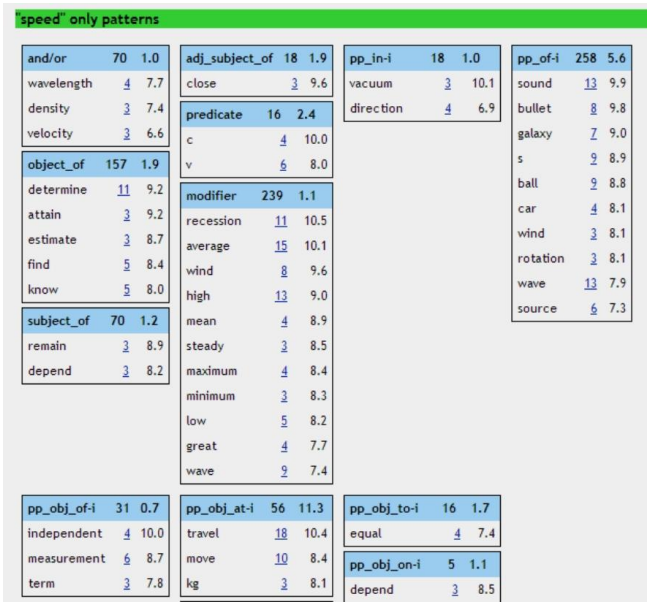


Figure 3. SPEED - Word Sketch

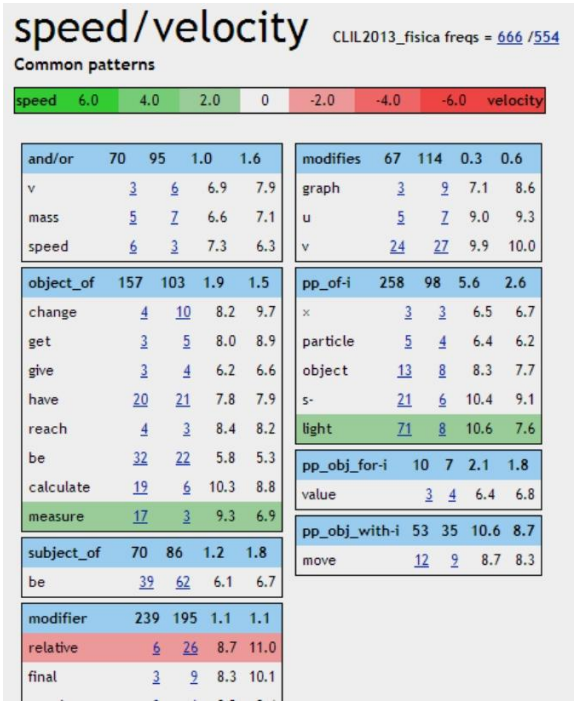


Figure 4. SPEED/VELOCITY - Common patterns

4.1 Case study: Bottom-up approach in *Ideal Gas Law*

Within a CLIL methodological course for inservice subject teachers given at the University of Turin in 2013, participants were introduced to corpus linguistic tools for teaching purposes. They were asked to work on disciplinary corpora created with the Sketch Engine and to reflect upon the language they should present to their students, creating a path for content and language integrated learning and teaching.

They first extracted the word list from their corpora, then they asked queries for LSP collocations, expanded the context of the occurrences to explore possible different meanings and finally created the word sketch of the keywords they thought to be crucial for content understanding. After a process of self-awareness language acquisition, they sketched the same - simplified and adapted - activities for their students with the aim to render the content accessible.

As an example the didactic unit about Ideal Gas Law[3] will be here analyzed. Corpus-based approach was used both to actively collect a LSP vocabulary and to give a warming up summary of the topics to be studied in depth throughout the unit.

At a preliminary stage the teacher makes a word list of nouns, verbs and adjectives in order to get a handle of the lexical material he/she is going to deal with, the he/she chooses the most significant items to be dealt with: *gas, temperature, volume, pressure, particle, collision, constant, proportional, universal, absolute*.

Starting from the first word on collocations are extracted and word sketches are drawn.

The most frequent attributes of the noun gas are ideal and real and it is often associated to the expressions *temperature of… / …at temperature*; *volume of… / …at volume*; *pressure of… / …at pressure*; *state of*…etc, and to the verbs *expand, compress, behave like,* besides occurring in the phrases *gas equation, gas law, gas state.*

From the disciplinary point of view, these occurrences introduce through expanded contextualized examples the differences between *ideal gases* and *real gases* and the physical quantities *temperature*, *volume* and *pressure,* which typify the state of gases.

As for these quantities students could be asked to fill in a table extracting information from collocations and word sketches, thus being actively involved in the bottom-up elaboration process.

	attributes	**subj./obj. of verbs**
temperature	thermodynamic high/low absolute constant proportional	increase/decrease rise keep measure depend
volume	small/large constant proportional	increase/decrease occupy keep measure depend
pressure	high/low constant proportional	increase/decrease exert keep measure

Some adjectives linked to *temperature* (*thermodynamic/absolute*) are part of the definition of the Kelvin temperature scale and of the concept of absolute zero; the verbs *keep* and *constant* are part of the occurrences *provided volume / temperature / pressure is kept constant*, which express Boyle's and Gay-Lussac's laws. The presence of *proportional* in connection to the three nouns suggests a relationship between all these quantities and it is frequently connected to the adverbs *directly* and *inversely*, the numerous examples at students' disposal also offer a linguistic model for expressing direct and inverse proportionality in English.

The syntagmatic relations of the keyword *particle* give some clues on the modality of interaction between the molecules of ideal gases: it occurs with the verbs *collide* and *interact*, in particular *interact by/ through/ on collision*, while *collision* has its highest frequency concordances with the adjectives *elastic /inelastic.* And so on.

Starting from the ten selected keywords this bottom-up approach allows students to get a sizeable portion of the LSP needed and to draw a fairly detailed mind map to scaffold further

[3] The Didactic Unit was experimented by professor Anna Grazia Botti

exercises such as cloze texts of reading comprehension tasks.

5 . CONCLUSIONS

CLIL teachers are confronted with a challenging task, which implies a clear mind about the features of the LSP they are dealing with. General dictionaries, CALL, machine translation tools are not enough to support them in handing out content through a foreign language.

Where traditional approaches show their limits, the integration of corpus-based approaches in disciplinary teaching and learning proves essential. On the one hand getting familiar with corpus analysis allows teachers to improve their own linguistic knowledge, on the other hand word sketches, collocations, frequency lists help them in selecting, planning and organizing didactic materials. Co-occurrences show which verbs are associated to a certain key-noun, which are the right prepositions or the most suitable adverbs, and their position. It is all about a knowledge that enriches the teachers' language in class and reinforces language awareness. The same happens with students who get involved in the process of knowledge construction and learn how to disambiguate polisemous terms and how to choose between near-synonyms inferencing linguistic information right from the context, thus - hopefully - avoiding to rely exclusively and rashly on automatic translation for reading comprehension and writing production.

References

Arispe, K (2014), *What's in a Bot? L2 Lexical Development Mediated Through ICALL*, Open Journal of Modern Linguistics 2014, 4: 150-165

Chun, D.M., Plass, J.L., (1996) *Effects of multimedia annotations on vocabulary acquisition*. The Modern Language Journal 80 (2): 183–198.

Coady, J., Magoto, J., Hubbard, P., Graney, J., & Mokhtari, K. (1993). High Frequency Vocabulary and Reading Proficiency in ESL Readers. In T. Huckin, M. Haynes, & J. Coady (Eds.), *Second Language Reading and Vocabulary Learning.* Ablex Publishing Corporation, Norwood, NJ: 217-228.

Coonan C. M. (2007) *Insider Views of the CLIL Class Through Teacher Self-observation–Introspection*, in International Journal of Bilingual Education and Bilingualism. Vol. 10: 625-646.

Durrant, P. (2009), *Investigating the viability of a collocation list for students of English for academic purposes*. English for Specific Purposes, 28: 157–169.

Firth, J. R. (1968). *Descriptive linguistics and the study of English.* In F. R. Palmer (Ed.), Selected papers of J.R. Firth 1952–1959. Longman, London: 96-113.

Gaskell, D. & Cobb, T. (2004), *Can learners use concordance feedback for writing errors?*, in System 32: 301-319.

Gavioli, L. (2005). *Exploring corpora for ESP learning*. Amsterdam: John Benjamins.

Gledhill, C. (2000). *The discourse function of collocation in research article introductions.* English for Specific Purposes, 19: 115–135.

Hüttner, Julia; Smit, Ute; Mehlmauer-Larcher, Barbara (2009), *ESP teacher education at the interface of theory and practice: Introducing a model of mediated corpus-based genre analysis*, in System 37: 99–109.

Kilgarriff, A. (2006), Collocationality (and how to measure it), in Proceedings of the XII EURALEX International Congress, Dell'Orso, Alessandria: 997-1005.

Louw, W. (1993), *Irony in the text or insincerity of the writer: The diagnostic potential of semantic prosodies.* In M. Baker, G. Francis, & E. Tognini-Bonelli (Eds.), Text and technology: In honour of John Sinclair. John Benjamins, Amsterdam: 157–176.

Marello, C. (2014), *Using Mobile Bilingual Dictionaries in an EFL Class*. In Proceedings of the XVI EURALEX International Congress. Bozen, Eurac Press: 63-84.

Mindt, D., 1997. *English corpus linguistics and the foreign-language teaching syllabus.* In: Thomas, J., Short, M.H. (Eds.), Using Corpora for Language Research: Studies in Honour of Geoffrey Leech. Longman, Harlow: 232–247.

Nelson, M. (2006). *Semantic associations in business English: A corpus-based analysis.* English for Specific Purposes, 25: 217–234.

Nerbonne, J. (2003). *Natural Language Processing in Computer-Assisted Language Learning*. In R. Mitkov (ed.), The Oxford Handbook of Computational Linguistics. Oxford University Press: 670-698.

McEnery, T., Xiao, R. (2010). *What corpora can offer in language teaching and learning*. In E. Hinkel (Ed.), *Handbook of Research in Second Language Teaching and Learning.* (Vol. 2). Routledge, London & New York: 364-380.

Richards, J.C., & Farrell, T.S.C. (2005), *Professional Development for Language Teachers*. Cambridge University Press, Cambridge.

Schmidt R. (1990), *The role of consciousness in second language learning*, in Applied linguistics 11:129-158.

Snow, C. E. (2010) *Academic language and the challenge of reading for learning about science*, in Science 328: 450–2.

Ting, T. Y.L. (2011), *CLIL... not only not immersion but also more than the sum of its parts*, in ELT Journal, May 2, 2011.

Tsui, A. (2003), *Understanding Expertise in Teaching*. Cambridge University Press, Cambridge.

Vannestál, M., & Lindquist, H. (2007). *Learning English grammar with a corpus: Experimenting with concordancing in a university grammar course, in* ReCALL, 9: 329–350.

Walker, C. 2011), *How a corpus-based study of the factors which influence collocation can help in the teaching of business English*, in English for Specific Purposes 30: 101–112.

Towards a collocation writing assistant for learners of Spanish

Margarita Alonso Ramos
Universidade da Coruña
Faculty of Philology
Campus da Zapateira s/n
15071 A Coruña (Spain)
lxalonso@udc.es

Marcos García Salido
Universidade da Coruña
Faculty of Philology
Campus da Zapateira s/n
15071 A Coruña (Spain)
marcos.garcias@udc.es

Orsolya Vincze
Universidade da Coruña
Faculty of Philology
Campus da Zapateira s/n
15071 A Coruña (Spain)
ovincze@udc.es

Abstract

This paper describes the process followed in creating a tool aimed at helping learners produce collocations in Spanish. First we present the *Diccionario de colocaciones del español* (DiCE), an online collocation dictionary, which represents the first stage of this process. The following section focuses on the potential user of a collocation learning tool: we examine the usability problems DiCE presents in this respect, and explore the actual learner needs through a learner corpus study of collocation errors. Next, we review how collocation production problems of English language learners can be solved using a variety of electronic tools devised for that language. Finally, taking all the above into account, we present a new tool aimed at assisting learners of Spanish in writing texts, with particular attention being paid to the use of collocations in this language.

1 Introduction

This paper[1] presents the process followed in developing a tool that helps learners of Spanish as L2 to produce collocations. Following Hausmann (1989), Mel'čuk (1998) and others, we assume that a collocation is a restricted binary co-occurrence of two lexical units (LUs) where one of them (the *base, B*) is chosen freely and the other (the *collocate, C*) is chosen idiosyncratically depending on B; cf., e.g., *take a walk, dar un paseo, faire une promenade*[2]. It has often been claimed that collocations are challenging for second language learners. In fact, the difference in collocational knowledge has been found to constitute an important factor that contributes to the difference between native and non-native language use (e.g. Howarth, 1998; Granger, 1998; Higueras García, 2006).

When producing a text, a language learner may face different types of problems relating to how words are combined in a native-like way. For instance, German learners of Spanish may wonder how to translate the collocation *einen Spaziergang machen* from their native language to Spanish, for which they need to know that in the case of this combination the verb *machen* translates to Spanish *dar* (lit. 'give'), and not *hacer* ('make'). This example shows a production problem. In other cases, learners may need information concerning the meaning of a collocation, for example, *sacar buenas notas* 'to get good grades'. Furthermore, the complexity of collocations is not limited to knowing which lexical item to combine with another, but it also concerns grammar. For instance, in order to avoid errors such as those found in the following learner sentence: *Los gays deben tener los derechos para casarse* (lit. 'Gays must have the rights in order to marry'), a learner of Spanish has to know not only that *derecho*

[2]Note that this definition does not use frequency of the combination as a determinative criterion, rather it emphasizes the lexical restriction imposed by one element on the selection of the other, in contrast with the approach promoted by corpus linguistics (Sinclair, 1991).

('right') goes with the verb *tener* ('to have'), but also that it is used in the singular form, without a determiner, and that it governs the preposition *a* (not *para*).

Given all these needs, we may raise the question of what the ideal resource designed to help learners overcome difficulties posed by collocations should be like. A straightforward answer would be the dictionary; however, we must be aware that in recent years the traditional dictionary format has been facing a serious crisis due to the challenges posed not only by online lexical and translation tools, but also by language corpora containing vast amounts of lexical information. Corpus-driven lexicography has given rise to what can be called "lexically-driven corpora", i.e. resources which do not provide lexical information in the form of a dictionary, but in the form of a concordance program exploiting language corpora. Through an appropriate user interface lexical items become pointers to the texts that reveal their meaning, blurring the boundaries between dictionaries and corpora (see Alonso Ramos, 2009). Some authors even claim that corpora can completely substitute dictionaries (e.g. Sinclair, 1987).

It is clear that the concept of the dictionary is changing towards a more flexible and dynamic tool, which aims to better address user needs, to the extent that certain authors propose alternative terms -e.g. leximat (Tarp, 2008) or lexical site (Jousse et. al. 2008)– to refer to this newly emerging concept. Jousse et al. (2008), in particular, argue that the word *dictionary* carries connotations of a linear structure, failing to describe the concept of a constantly evolving network, embodied by modern online lexical tools and constituting a better model of lexical knowledge. Independently of the term we use to refer to these new lexical resources, the fact is that dictionaries have ceased to be stand-alone products, which means that they are increasingly integrated with other resources such as corpora, other dictionaries, and glossaries. They also serve to complement and are in turn well complemented by CALL applications.

What we have described so far matches the course of the evolution taken by our research interests detailed in this paper: from an online collocation dictionary of Spanish (DiCE), the development of which began ten years ago, towards an online collocation writing assistant, integrated with the DiCE. In the next section, we briefly present the DiCE and explain the motivations behind the development of a further tool that would complement it. Section 3 focuses on the potential user of a collocation learning tool, examining the usability problems posed by the DiCE and exploring language learners' needs through a learner corpus study of collocation errors. As we will show, both of these aspects should be taken into account when designing a collocation writing assistant. Section 4 provides an overview of freely available online lexical tools for English that can potentially resolve collocation production problems. Section 5 describes in detail the architecture of a new tool aimed at assisting Spanish as L2 learners' collocation production. Finally, Section 6 draws some conclusions from the work presented here and outlines the direction of future research in the area of automatic collocation error detection and correction.

2 Starting from an online collocation dictionary

The *Diccionario de Colocaciones del Español* (DiCE), a web-based collocation dictionary of Spanish, has been available online since 2004, its database constantly being improved and expanded. Since the dictionary has been described in detail on various occasions (e.g. Alonso Ramos, 2005; 2006; 2008; Alonso Ramos et al. 2010a), here we only provide a brief presentation of its main features and focus on the reasons for developing a further tool that enables some of its drawbacks to be overcome.

The DiCE constitutes an online implementation of the principles of lexical description proposed by the Explanatory Combinatorial Lexicology (ECL, Mel'čuk et al.,1995). In addition to providing a theoretically well-founded description of collocations, it aims to be a useful tool not only for specialized researchers but also for the general public. To this end, lexical functions, the formal representation used to describe the semantic and syntactic features of collocations, are paraphrased in natural language glosses. At the same time, the web interface has been designed

to enable flexible access to the electronic lexical database, with a view to satisfying the needs of a broad range of users, from researchers through language learners to lexicographers working on DiCE.

In accordance with our framework, we conceive of collocations as restricted combinations of two lexical units, the base and the collocate. For instance, in the combination *reanudar una amistad* 'renew a friendship', the noun is the base, and it conditions the selection of the collocate verb.

The user interface of the DiCE consists of three main components: 1) the *dictionary* itself, 2) the *advanced search component*, and 3) the *learning module*. The *dictionary component* provides access to the contents in a way similar to other collocation dictionaries. Users are offered a list of lemmas, each associated with its lexical units, under which corresponding semantic and combinatorial information can be found.

In order to offer dynamic access to the information stored in the DiCE database, the advanced search component offers four options. Each of these was designed to provide the user with a more direct path of access to a specific type of information:

a) *What does it mean?*: a reception oriented module providing direct access to the entry of a specific collocation. The user is expected to introduce a base (e.g. *amistad*) and a collocate (e.g. *reanudar*) to be directed to the entry of the corresponding collocation.

b) *Writing aid*: a production oriented module, which allows the user to find collocates of a given base (e.g. *amor* 'love'), corresponding to a specific part of speech and meaning (e.g. 'felt for one another'), such as *amor mutuo* 'mutual love'.

c) *Direct search*: an option which serves to find collocations encoded by a specific Lexical Function (e.g. Sing(remordimiento) = *acceso de ˜* 'fit of remorse').

d) Inverse search: a module where the user is asked to introduce a collocate (e.g. *cumplir* 'fulfill') in order to find the bases it can be combined with (e.g. *deseo* 'wish', *esperanza* 'expectation').

Finally, the third component, the *learning module*, aims to provide the user with learning material concentrating on collocations. For the present it is limited to a few sections containing exercises related to a particular topic, one of which is an introduction to the use of the DiCE itself.

However, these learning activities do not differ consistently from those available on paper, but, just as an e-dictionary should offer more advanced features rather than being a mere electronic version of a paper dictionary, e-learning activities should be different from traditional teaching material. First of all, the collocation verification process should enable the user to access external language corpora, besides relying on the dictionary's own database. For instance, if in an exercise aimed at practising intensifier collocates, a learner provides *total* 'complete' as a collocate of *admiración* 'admiration', the current system will treat it as incorrect because this combination is not included in the DiCE database. However, a search in external corpora would enable the user to check whether the collocation is used in language and with what frequency as compared to other combinations with a similar meaning.

The use of language corpora is being promoted in language teaching since it is in line with the current trend of emphasizing autonomous learning. We also had the idea that learner autonomy could be further reinforced by the creation of a learning space in which learners can administer their personal collocation dictionaries, annotations, performance scores and problems identified in relation to specific collocations or collocation types. Ultimately, we believed that an ideal CALL environment focusing on collocations should tightly integrate a number of different components: a collocation database, a corpus interface, a collocation checker tool and other learning utilities, in order to support the users' collocation production in writing tasks.

These ideas constituted the main incentive behind the development of an interactive collocation learning environment. In order to create such tool, it was necessary to learn about its potential users, to which end we set out to gather information on users' reference skills when it comes to using a collocation database such as the DiCE, as well as on language learners' collocation proficiency. In the following section we will briefly present some findings concerning these two aspects.

3 Getting to know the user

3.1 Users' reference skills

As claimed above, the modifications of the DiCE interface were aimed at turning it into a useful tool for a wide range of users. This is the reason why a usability test was carried out to see how well different target user groups were able to perform with the dictionary. The aim of the test was to assess the different search options offered by the interface both in terms of efficiency and the adequacy of the layout, as well as to examine whether users' reference skills met those required by the DiCE.

In relation to user skills and preferences, the study, described in detail in Vincze and Alonso (2013), revealed that subjects were rather reluctant to explore the dictionary interface in search of different search options and that they were not familiar with certain terms applied in the dictionary. It was observed that subjects preferred to stick to familiar or more straightforwardly accessible search options, and did not show willingness to experiment with unknown or more novel functions. This could be seen in that they most frequently used the *Dictionary module* instead of more specific search options that could have provided more direct and quicker access to the items they were required to look up. The reason for this could be, on the one hand, that this access path is offered by default in the web interface, and, in addition, it allows the correct answer to be retrieved in the case of most questionnaire items; consequently when participants managed to find the required information in this way, they did not turn to the advanced search options. Furthermore, the type of access provided by this module is very similar to paper dictionaries and may therefore seem more familiar to users. Another finding pointing to the direction of users' preference for familiar search options was that the second most frequently and most successfully used query type was *What does it mean?*. It can be argued that this query type stands for the most common type of dictionary use, i.e. looking up a given lexical item in order to check its meaning or its spelling, as opposed to production oriented look-ups represented by the *Writing aid* option.

With respect to participants' reference skills, it was found that a lack of knowledge concerning the terminology applied in the dictionary caused difficulties in interpreting the dictionary content involving some of the query interfaces and the presentation of lexicographic data. Subjects were often unfamiliar with the notion of collocation and the specific terminology applied in the DiCE, leading them to confuse the elements of collocations (the base and the collocate), as well as with the more general concepts of word form and lemma, complicating the use of a number of search options.

In conclusion, the usability study of the DiCE interface showed that potential users of an online lexical learning environment 1) are more used to manipulating lexical resources in reception than in production tasks, and that 2) they might be more successful at using a tool whose functions do not differ radically from resources they are already familiar with, 3) whose search options are not highly modular, and 4) which keeps reference skill requirements to the minimum.

3.2 Language learners' collocation use

In order to design useful learning tools, it is necessary to know how learners use collocations. Previous studies (Alonso Ramos et al. 2010b, 2010c; Vincze et al., 2011; Wanner et al., 2013a), addressed the following two research questions for Spanish as L2: (1) Can errors in learners' collocation use be systematized? (2) How can this systematization be exploited in CALL and, more specifically, in active CALL-based collocation learning, to offer the learner not only a list of possible corrections, but also concrete correction suggestions and learning material targeting the type of error?

Previous work suggests that a CALL environment focusing on collocations can profit from data on learners' actual language behaviour obtained from corpus research (Shei and Pain, 2000; Chang et al., 2008). In order to gain information on the collocation knowledge and typical errors of Spanish as L2 learners, correct and erroneous collocations in a portion of the CEDEL2 corpus[3] (Lozano and Mendikoetxea, 2013) were

[3]CEDEL2 is an L1 English-L2 Spanish learner corpus containing essays written by English mother tongue Spanish L2 learners see `http://www.uam.es/`

annotated. Although currently available general learner error typologies tend to group collocation errors into a single subclass of lexical errors (Aldabe et al., 2005; Miličevič and Hamel, 2007; Granger, 2007; Díaz-Negrillo and García-Cumbreras 2007), a closer look at the learner corpus revealed that a considerably more detailed collocation error typology is needed in order to offer more targeted (and thus more effective) learning exercises, and to facilitate the development of techniques for automatic correction of collocation errors in learner writing.

Consequently, we created a detailed collocation error typology, which distinguishes three parallel dimensions (for a more detailed description see Alonso Ramos et al., 2010b and 2010c). The first of these captures the location of the error, i.e. whether it affects the base, the collocate, or the collocation as a whole. The second dimension models descriptive error analysis and distinguishes between three main types of error: lexical, grammatical and register error. Finally, the third dimension represents explanatory error analysis: it classifies errors according to their perceived source into one of the two main categories of transfer errors, namely errors reflecting L1 interference or interlanguage errors, the result of incomplete knowledge of the L2 without L1 interference.

The annotated corpus contains 46,266 words, in which a total number of 1938 collocation tokens, corresponding to 1171 collocation types were identified during the manual annotation process. Manual selection of collocations was necessary since our aim was to only examine combinations which qualify as collocations following our theoretical framework (see Section 1). Out of the total number of annotated collocation tokens, 1481 are correct and 457 are erroneous.

As for the location dimension, it was found that lexical errors most often affect the collocate, in a total of 180 collocations (62%), see (1), although a relatively large proportion, 62 collocations (21%) have erroneous bases, see (2), with cases of collocations having both an incorrect base and collocate, see (3), while 50 expressions (17%) contain a lexical error that is considered to affect the collocation as a whole. These results suggest that a genuinely effective CALL system should not be limited to recognizing errors in the collocate, as in e.g. Liu (2002) or Chang et al. (2008) (see below), but should also foresee lexical errors concerning the base or even both elements of the collocation.

(1) **interrumpir una regla* 'interrupt a rule' instead of *romper una regla* 'break a rule'

(2) **lograr un gol* 'achieve a goal (in sport)' instead of *lograr un objetivo* 'achieve an aim

(3) **pasar un testemuño* 'pass a testimony (from Portuguese)' instead of *dar testimonio* 'give testimony'

Automatic correction of the third error type included in the location dimension may present a considerable challenge. Errors affecting the collocation as a whole include incorrect collocation-like expressions that should be correctly expressed by a single word (4) and cases of incorrect single-word forms used instead of a collocation (5)

(4) **poner apasionado* 'make passionate' instead of *apasionar* 'to fascinate'

(5) **misenterpretación* 'misinterpretation' instead of *mala interpretación*

With respect to the explanatory error type dimension, of the 292 lexical collocation errors found in the corpus (note that a collocation can contain more than one error), 60% were labeled as transfer errors, while 40% were annotated as interlanguage errors. This is in line with the findings of other authors such as Liu (2002), Nesselhauf (2005), etc. Our corpus data also corroborates the hypothesis that in most lexical collocation errors, the erroneous element can be conceived of as a synonym or a translation synonym of its correct counterpart for correction purposes, a feature that can be made use of by automatic tools (Liu, 2002; Chang et al., 2008; Futagi, 2010). Remarkably, our data shows this to be true both in the case of L1 transfer and interlanguage errors. Nevertheless a small number of error types do not fit into this picture.

Errors resulting from the phenomenon commonly known by language learners and teachers

proyectoinv/woslac/cedel2.htm.

as 'false friends (6) constitute such a case. Similarly, in the case of errors involving the use of lexical elements which constitute non-words in the target language (7), using translation equivalents or synonyms to provide correction suggestions may be problematic and/or insufficient. Here, the introduction of a strategy involving edit-distance should be considered.

(6) Hemos **licenciado en el colegio* (from college) en la vecina ciudad Lit. We earned a degree in the primary school in the neighbor town

(7) En Oaxaca se puede **ir de hiking* (instead of *hacer senderismo*) Lit. In Oaxaca one can go hiking

In addition to lexical errors, learner tools aimed at the correction of collocations should also take grammatical errors into account. From our point of view, certain grammatical errors are to be considered proper collocation errors, due to the fact that they affect the correct formulation of a lexical combination. In fact, grammatical collocation errors (see (8), (9) and (10)) were found rather frequently in the corpus, concerning 198 (45%) of the 457 erroneous collocations annotated.

(8) determination error: **tomar sol* instead of *tomar el sol* 'to sunbathe;

(9) incorrect government: **montar a bicicleta* instead of *montar en bicicleta* 'to ride a bike; *asisto la Universidad* instead of *asisto a la Universidad* 'I attend the university;

(10) incorrect number: **estamos en vacación* instead of *estamos de vacaciones* we are on holiday.

As we have shown in this section, learner errors affecting collocations can be of many kinds, and can be systematized in a specific typology. A sufficiently fine-grained distinction of error types can not only provide useful input for the design of teaching material, but can also be made use of when determining the strategies to be implemented in a tool offering automatic correction suggestions for collocation errors. Once we have a clearer idea of the difficulties learners have to face at the moment of using a collocation learning tool, as well as of the diversity of collocation errors made by learners of Spanish as L2, we can go on to examine some existing lexical tools for learners of English in order to verify whether they can solve some of the problems posed by collocations.

4 Facing the difficulties of writing texts through the use of online lexical tools

When producing a text in English, learners have at their disposal a number of online tools that help them cope with some of the problems described above. In this section, we examine a number of these tools, since, to the best of our knowledge, there are no resources of this kind for learners of Spanish. Depending on the type of information sought by learners and the output these resources produce, we have classified them into three groups, the first of which includes those tools that in some respects resemble conventional combinatorial dictionaries; in the case of the second group, the query interface is similar to that found in an electronic dictionary, but the output consists roughly of n-grams or strings of word forms; and finally, the third group consists of tools that enable users to verify whether a combination produced by them is correct or not.

Dictionary-like tools. If a learner is interested in finding out about the combinatorial properties of already known lexical units, they may use a collocation dictionary or tools such as the *Learning collocations* component of FLAX[4] (Wu et al., 2010), the automatic collocation dictionary *For better English*[5] or the *Combinations* utility of *Just the word*[6]. When using these tools, in much the same way as with a collocation dictionary, users look up the word they are interested in, and obtain its collocates sorted according to their syntactic structure (e.g. V+N, Adj+N, etc.). In one case (*Just the word*), the collocations are also grouped according to semantic proximity. Additionally, *Learning collocations* and *Just the word* provide frequency information for each collocation.

[4] http://flax.nzdl.org/greenstone3/flax?a=fp&sa=collAbout&c=collocations&if=flax

[5] http://forbetterenglish.com/

[6] http://www.just-the-word.com/

The way the user accesses a collocation dictionary like the DiCE is very similar, since, as explained above, the *Dictionary Module* provides access to collocates by looking up a lemma. Likewise, the information provided by the DiCE (syntactic structure, semantic grouping, frequency of the collocation) is as complete as that offered by the tools examined. With some of these tools, however, users' access to corpus information is more direct, since it is not filtered by the lexicographer's criterion. In addition to this, one of the tools examined (*Learning collocations*) offers the possibility of picking examples from corpora and storing them in the users' personal dictionary.

String-searching tools. Like the previous ones, tools of this kind can be used to obtain information about the combinations of a certain word or phrase. Their output, however, is less refined than that of a collocation-searching utility, since it lists strings of all kinds in which the target word or phrase is found. If users want to narrow down their search because they are only interested, for instance, in finding occurrences of the target word as the object of a certain verb, they can refine their query by specifying certain categorial or distributional features. Thus, the *Lexcheker* of *StringNet*[7] (Wible et al., 2011) allows its users to exploit different degrees of specification by combining word class information and word-forms (e.g. [verb] *step*), whilst in the *Web Phrases* component of FLAX users can specify the distribution and length of the strings that combine with the target word or phrase.

Besides providing information about the correctness or the frequency of a particular combination, these tools can be especially useful for raising learners' awareness about grammatical restrictions related to the combination at hand (e.g. whether a certain verb takes a to+infinitive complement or gerund; preposition selection, etc.).

Collocation checkers. By means of the resources examined so far, a learner aiming to use a certain lexical item and wanting to know which other words can be combined with it can find the correct word choices and discard incorrect ones. With a collocation checker, however, learners who have already come up with a certain combination that they believe expresses the meaning they want to convey can seek a confirmation or a rejection of their hypothesis. Tools such as the *Collocation checker*[8] (Chang et al., 2008) or *Just the word* (when searching for a phrase instead of a single word) can be employed to this end, since they provide the user with feedback concerning the correctness of the combination introduced (based on its attestation in corpora) together with frequency information and suggestions of other possible combinations.

Some limitations of this type of tools have to do with the (lack of) coverage of all possible types of learner errors. The *Collocation checker*, for instance, focuses on V+N collocations and gives feedback on whether a verb can be combined with a certain noun. Thus, if the collocation proposed by the learners is attested in corpora, they will receive a message stating its correctness and a list of related constructions. If the verb does not occur with the noun, the application will indicate either that the collocation "might not be appropriate" or that it does not recognize such an expression and will provide alternatives with other verbs. However, as shown above, collocation errors can affect different parts of a combination. Thus, if we search for a combination of a verb plus a non-existent noun (e.g. **make cite*, instead of *make an appointment*, cf. Sp. *cita* 'appointment'), the tool will not provide any useful feedback to our query. Besides, the feedback given to infelicitous searches contains linguistic or lexicographic terminology (e.g. *lemma, support verb*) that may be unfamiliar to users, as the DiCE usability test has suggested.

After having observed some tools that help learners find or check collocations, the following section presents a collocation learning assistant for learners of Spanish.

5 Getting closer to a collocation writing assistant

As already pointed out above, collocation errors can be of different types and degrees of complexity. As stated in Wanner et al. (2013b), the differing complexity of collocation errors has further consequences for the prospects of successful au-

[7] http://www.lexchecker.org/

[8] http://miscollocation-richtrf.rhcloud.com/

tomatic recognition and correction in case of erroneous use: some of them will be more easily and more accurately recognized and corrected by state of the art techniques than others, whilst some of them require a further step to be taken. In what follows, we first introduce the requirements for a collocation checker tool, after which we provide a brief presentation of the HaRenEs[9] interface under development, a learning tool focusing on Spanish collocations[10].

5.1 Requirements for a collocation writing assistant

On the basis of the conclusions drawn from the usability and learner corpus studies previously presented, as well as the overview of existing online lexical tools provided, it is possible to formulate a list of requirements for the learning environment we aim to create. These can be organized in the following way:

- The target of the learning tool: the proposed tool should focus on collocations as understood within our theoretical framework (see Section 1). This means that we do not wish to treat phraseological strings that are produced as non-compositional chunks, such as *de acuerdo con* 'in accordance with'. We will concentrate strictly on restricted lexical co-occurrence phenomena, as in e.g. *acuerdo tácito* 'tacit agreement'[11].

- Accuracy of correction: the learning tool must in all cases provide feedback regarding the correctness of a collocation introduced, and, in the case of incorrect combinations, it should provide accurate correction suggestions. By this we mean that the collocation checker has to determine the nature of the error, including grammatical errors (e.g. **asistir la universidad* '*assist university*').

- Integration with other resources: the learner tool should be integrated with corpora and dictionaries. All suggested collocations should be illustrated with corpus examples, and the user should be redirected to existing entries in the DiCE or other online dictionaries.

- Features supporting usability and learning: users should have at their disposal a personalized collocation dictionary in which they can include new collocations accompanied by examples, as well as collocation errors. Collocation look-up and checking should be available by introducing either a stand-alone collocation or a text. When the interface is used to verify collocations in running text, the user should be able to further edit the text once it has been verified. Dictionary look-ups should be available both through the syntactic pattern and the semantic content of a collocation. Users should be provided with a number of learning activities for practicing collocations learnt through the collocation checker (similarly to FLAX).

5.2 HaRenEs Writing Assistant

The HaRenEs Writing Assistant is currently being developed in a joint project at the University of A Coruña and Pompeu Fabra University. The current learning environment consists of three main components: 1) the collocation checker, 2) the collocation search and 3) the personal dictionary. The collocation checker allows users to verify the correctness of a specific Spanish collocation and, in the case of incorrect combinations, to request correction suggestions, as well as usage examples of a given collocation in context. Users can introduce a single collocation in the search box, not necessarily in the lemma form (e.g *dimos un paseo* 'we took a walk'); and they can also request the verification of collocations in running text. Figure 1 shows a screenshot of the HaRenEs interface in use.

[9] HaRenEs stands for "Herramienta de Ayuda a la Redaccin en Español: Procesamiento de Colocaciones".

[10] A demo version of the HaRenEs interface can be seen at: `http://harenes.taln.upf.edu/CakeHARenEs`

[11] We are aware of the fact that a sharp distinction cannot always be drawn between full idioms and collocations. However, we believe that the learning of these two types of multiword units differs considerably: among other things, full idioms are difficult to understand, but collocations are difficult to produce. The learner needs to know the collocation *acuerdo tácito* to speak about a kind of agreement, i.e. one that is implicit, not overtly expressed. On the contrary, *de acuerdo con* is learnt as a whole string since it does not contain the meaning 'acuerdo', but expresses a completely different meaning: [X] de acuerdo con Y: '[X] following the rule or the system Y or Y's wishes'.

Figure 1: The HaRenEs user interface

Unlike other proposals, our checker will offer accurate corrections of collocation errors, rather than lists of possible combinations ranked according to frequency. Furthermore, the system provides the option of linking any frequent learner error to the personal dictionary. Even though the different identification techniques used by the collocation checker are still in development (Ferraro et al., 2011; Moreno et al., 2013; Wanner et al., 2013b; Ferraro et al., 2014), the results obtained so far are promising. The system is being trained with data from CEDEL2. In Table 1 we provide examples of learner errors found in the corpus together with the corrections automatically suggested by the tool (see Ferraro et al., 2014).

Error	**Suggested Correction**
realizar meta lit. 'to realize an aim'	*alcanzar una meta* 'achieve an aim'
cambiar al cristianismo 'to change to Christianity'	*convertirse al cristianismo* 'to convert to Chistianity'
concluir un problema 'to conclude a problem'	*resolver* 'solve a problem'

Table 1: Suggested corrections of collocation error provided by HaRenEs

In order to verify the effectiveness of the collocation checker with running text, we carried out a test with full sentences taken from the learner corpus. For instance:

(11) *La hija está tratando de *capturar la atención de su madre*
lit. 'The daughter intends to capture the attention of her mother.'

In this case, the checker tool detects the incorrect collocation **capturar la atención* lit. 'capture the attention' and proposes *llamar la atención* lit. 'call the attention'. The interface allows the user to accept or reject each of the multiple suggestions, consult examples of the suggested collocation, add it as a new entry to the personal dictionary, and link the collocation error to an existing dictionary entry.

The second component, *Collocation search*, is also still under development. It is designed to be similar to the dictionary-like lexical tools using corpora introduced in Section 4. However, in contrast to these, our goal is not only to provide access to collocations via their syntactic pattern (e.g. verb+*miedo* 'fear' or *miedo*+adj), but also through a semantic typology. For instance, if a user is searching for a way to express the meaning related to the starting phase of fear, it would be desirable to find verb+object collocacions such as *coger miedo* 'take fear of sg', as well as subject+verb collocations like *entrarle miedo* 'fear enters sb', *asaltarle miedo* 'fear assaults sb', or *invadirle el miedo* 'fear invades sb'. Note that in existing lexical resources these combinations are normally not found in the same category, since they are classified according to syntactic pattern.

Concerning the third component, the personal dictionary, we believe that it is highly useful to provide the option of linking erroneous collocations with their correct counterparts. Similarly to FLAX, users can be given the option of creating and organizing collocation lists at will. In our case, however, by default each collocation included in the personal dictionary by a user will be automatically registered in an entry with a standardized structure including the following fields: base, collocate, syntactic pattern, semantic class, examples and observations.

Unlike some of the other tools presented in Section 4, we do not allow the use of wild card operators in queries, since we try to keep user interactions as simple as possible for the sake of usability. Another point of difference with other lexical tools is that HaRenEs focuses on collocations, not on government: no direct queries can be carried out to find the preposition governed by

a given verb (e.g. *depender de* 'to depend on'). However, information on government that concerns a given collocation can be found. For instance, if a user wants to know whether a collocation such as *sentir miedo* 'feel fear' governs the preposition *a* or *de*, they can find this information in the examples coming from the corpus and also in the dictionary component.

An approach similar to that of StringNet would also be possible to implement, given that our reference corpus is tagged. However, before implementing this functionality, we need to test its efficiency with users. As we have seen in the usability test of the DiCE interface, we cannot take users' knowledge of technical linguistic terms or notions, such as e.g. names of parts of speech, for granted. And, ultimately, as mentioned above, the target of the HaRenEs environment is constituted by collocations, not merely frequent lexical combinations. However, although the metrics behind our tool are based on lexical frequencies, as is the case with other lexical checkers, we have set ourselves the challenge of automatically distinguishing between phraseological combinations such as *de acuerdo con* 'in accordance with' and genuine collocations such as *un acuerdo tácito* 'tacit agreement'.

6 Conclusions

Genuine lexical writing assistants that attempt to detect collocation errors have much less tradition in CALL than spelling and grammar checkers. In general they are not as mature as the latter: many of them are not successful enough in recognizing and correcting errors. However, this is not only due to the immaturity of the technologies. As we have shown, collocation errors are very heterogeneous and thus rather difficult to deal with.

Furthermore, the challenge not only lies in developing techniques capable of identifying and correcting collocation errors in a sufficiently accurate and efficient way, but also in designing an interface which any L2 learner can manipulate with ease. As pointed out above, there is a general tendency to blur the boundaries between dictionary and corpus and, going even further, to make the lexical tool itself almost invisible to the user, hoping that the user will be able to find any desired answer with a single click of the mouse. This design strategy is already operational but only in the case of language comprehension, not for production purposes. We would like to draw attention to this important difference and to make an appeal for a concerted effort to be made to build an efficient writing assistant.

Acknowledgments

The work presented in this paper has been supported by the Spanish Ministry of Economy and Competitiveness (MINECO) and the EFRD Funds of the European Commission under the contract number FFI2011-30219-C02-01, as well as the Spanish Ministry of Education under the FPU grant AP2010-4334 and the Galician Government under the post-doctoral grant POS-A/2013/191.

References

Aldabe, I., B. Arrieta, A. Díaz De Ilarraza, M. Maritxalar, M. Oronoz and L. Uria. 2005. Propuesta de una clasificación general y dinámica para la definición de errores. *Revista de Psicodidáctica*, 10/2: 47-60.

Alonso Ramos, M. 2005. Semantic Description of Collocations in a Lexical Database. In F. Kiefer et al. (eds.), *Papers in Computational Lexicography COMPLEX 2005*. Budapest: Linguistics Institute and Hungarian Academy of Sciences, 17-27.

Alonso Ramos, M. 2006. Towards a Dynamic Way to Learn Collocations in a Second Language. In Corino, E., C. Marello and C. Onesti (eds.), *Proceedings of the Twelfth EURALEX International Congress*. Accademia della Crusca, Università di Torino, Edizioni dellOrso Alessandria, Torino: 909-923.

Alonso Ramos, M. 2008. Papel de los diccionarios de colocaciones en la enseanza de español como L2. In Bernal, E. and J. De Cesaris (eds.), *Proceedings of the XIII EURALEX International Congress*. IULA, Documenta Universitaria, Barcelona: 1215-1230.

Alonso Ramos, M. 2009. Hacia un nuevo recurso léxico ¿fusión entre corpus y diccionario? In Cantos Gómez, P. and A. Sánchez Pérez (eds.), *A Survey of Corpus-based Research. Panorama de investigaciones basadas en corpus*. AELINCO, Murcia: 1191-1207.

Alonso Ramos, M., A. Nishikawa and O. Vincze. 2010a. DiCE in the web: An online Spanish collocation dictionary In S. Granger, M. Paquot

(eds.), *eLexicograpy in the 21st century: New Challenges, New Applications. Proceedings of eLex 2009*. Cahiers du Cental 7, Presses universitaires de Louvain, Louvain-la-Neuve: 367-368.

Alonso Ramos, M., L. Wanner, N. Vázquez, O. Vincze, E. Mosqueira and S. Prieto. 2010b. Tagging collocations for learners. In S. Granger, M. Paquot (eds.), *eLexicograpy in the 21st century: New Challenges, New Applications. Proceedings of eLex 2009*. Cahiers du Cental 7, Presses universitaires de Louvain, Louvain-la-Neuve: 369-374.

Alonso Ramos, M., L. Wanner, O. Vincze, G. Casamayor, N. Vázquez, E. Mosqueira and S. Prieto 2010c. Towards a Motivated Annotation Schema of Collocation Errors in Learner Corpora. *7th International Conference on Language Resources and Evaluation (LREC)* La Valetta, Malta: 3209-3214.

Chang, Y.C., J. S. Chang, H.J. Chen, and H.C. Liou. 2008. An Automatic Collocation Writing Assistant for Taiwanese EFL Learners. A case of Corpus Based NLP technology. *Computer Assisted Language Learning*, 21(3):283-299.

Díaz-Negrillo, A. and M. A. García-Cumbreras. 2007. A tagging tool for error analysis on learner corpora. *ICAME Journal*, 31/1: 197-203.

Ferraro, G., R. Nazar and L. Wanner. 2011. Collocations: A Challenge in Computer-Assisted Language Learning. In I. Boguslavsky, L. Wanner (eds), *Proceedings of the 5th International Conference on Meaning-Text Theory (Barcelona, September 8-9, 2011)*: 69-79.

Ferraro, G., R. Nazar, M. Alonso Ramos and L. Wanner. 2014. Towards advanced collocation error correction in Spanish learner corpora. *Language Resources and Evaluation*, 48 (1): 45-64.

Futagi, Y. 2010. The effects of learner errors on the development of a collocation detection tool. In *AND'10 Proceedings of the fourth workshop on Analytics for noisy unstructured text data*. ACM, New York: 27-34.

Granger, S. 1998. Prefabricated patterns in advanced EFL writing: Collocations and formulae. In A. Cowie (ed.), *Phraseology: Theory, Analysis and Applications*. Oxford University Press, Oxford: 145-160.

Granger, S. 2007. Corpus d'apprenants, annotation d'erreurs et ALAO: une synergie prometteuse. *Cahiers de lexicologie*, 91/2: 465-480.

Hausmann, F. J. 1989. Le dictionnaire de collocations. In F. J. Hausmann et al. (eds.) *Wörterbücher-Dictionaries-Dictionnaires*, vol. 1. Gruyter, Berlin: 1010-1019.

Higueras García, M. 2006. *Las colocaciones y su ensenñanza en la clase de ELE*. Arco Libros, Madrid.

Howarth, P. 1998. The phraseology of learners' academic writing. In A.P. Cowie (ed.) *Phraseology. Theory, Analysis, and Applications*. Oxford University Press, Oxford: 161-186.

Jousse, A. L., A. Polguere and O. Tremblay. 2008. Du dictionnaire au site lexical pour l'enseignement/apprentissage du vocabulaire. In Grossmann, F. and S. Plane (eds), *Les apprentissages lexicaux. Lexique et production verbale*. Presses universitaires du Septentrion, Villeneuve dAscq : 141157.

Knublauch, H., R. W. Fergerson, N. F. Noy and M. A. Musen. 2004. *The Protégé OWL Plugin: An Open Development Environment for Semantic Web Applications. Third International Semantic Web Conference, Hiroshima, Japan*.

Liu, L. E. 2002. *A corpus-based lexical semantic investigation of verb-noun miscollocations in Taiwan learners' English*. Masters thesis, Tamkang University, Taipei.

Liu, A. Li-E., D. Wible, and N.-L. Tsao 2009. Automated suggestions for miscollocations. In *Proceedings of the NAACL HLT Workshop on Innovative Use of NLP for Building Educational Applications*. Boulder, CO: 47-50.

Lozano, C., and A. Mendikoetxea. 2013. Learner corpora and Second Language Acquisition: The design and collection of CEDEL2. In A. Díaz-Negrillo, N. Ballier and P. Thompson (eds.), *Automatic Treatment and Analysis of Learner Corpus Data*. John Benjamins, Amsterdam: 65-100.

Mel'čuk, I. A. 1996. Lexical Functions: A tool for the description of lexical relations in the lexicon. In L. Wanner (ed.), *Lexical Functions in Lexicography and Natural Language Processing*. John Benjamins, Amsterdam: 37-102.

Mel'čuk, I. A. 1998. Collocations and Lexical Functions. In P. Cowie (ed.), *Phraseology. Theory, Analysis, and Applications*. Clarendon Press, London: 23-53.

Melčuk, I., A. Clas and A, Polguère. 1995. *Introduction à la lexicologie explicative et combinatoire*. AUPELF-UREF/Duculot, Louvain-la-Neuve.

Miličevič, J., and M.-J. Hamel. 2007. Un dictionnaire de reformulation pour apprenants avancés du franais langue seconde. *Revue de lUniversité Moncton*, numéro hors série: 145167.

Moreno, P., G. Ferraro and L. Wanner. 2013. Can we determine the semantics of collocations without semantics?. In Kosem, I., Kallas, J., Gantar, P., Krek, S., Langemets, M., Tuulik, M. (eds.), *Electronic lexicography in the 21st century: thinking outside the paper*. Proceedings of the eLex 2013 conference, 17-19 October 2013, Tallinn, Estonia.

Trojina, Institute for Applied Slovene Studies/Eesti Keele Instituut, Ljubljana/Tallinn: 106-121.

Nesselhauf, N. 2005. *Collocations in a Learner Corpus*. Benjamins, Amsterdam.

Park, T., E. Lank, P. Poupart, and M. Terry. 2008. Is the sky pure today? AwkChecker: An assistive tool for detecting and correcting errors. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology (UIST '08)*. New York.

Shei, C.C. and H. Pain. 2000. An ESL writer's collocational aid. *Computer Assisted Language Learning*, 13(2):167182.

Sinclair, J. M. 1987. The Dictionary of the Future. Collins English Dictionary Annual Lecture. University of Strathclyde, 6 May 1987.

Sinclair, J. M. 1991. *Corpus, concordance, collocation* . Oxford University Press.

Tarp, S. 2008. *Lexicography in the Borderland between Knowledge and Non-Knowledge*. Niemeyer, Tübingen.

Vincze, O., M. Alonso Ramos, E. Mosqueira Suárez and S. Prieto González. 2011. Exploiting a learner corpus for the development of a CALL environment for learning Spanish collocations. In Kosem, I. and K. Kosem (eds.), Electronic lexicography in the 21st century: New applications for new users. *Proceedings of eLex 2011*. Institute for Applied Slovene Studies.

Vincze, O. and M. Alonso Ramos. 2013. Testing an electronic collocation dictionary interface: Diccionario de Colocaciones del Español. In Kosem, I., Kallas, J., Gantar, P., Krek, S., Langemets, M., Tuulik, M. (eds.), *Electronic lexicography in the 21st century: thinking outside the paper. Proceedings of the eLex 2013 conference, 17-19 October 2013, Tallinn, Estonia*. Institute for Applied Slovene Studies, Trojina/Eesti Keele Instituut, Ljubljana/Tallinn: 328-337.

Wanner, L., M. Alonso Ramos, O.Vincze, R. Nazar, G. Ferraro, E., Mosqueira, S. Prieto. 2013a. Annotation of collocations in a learner corpus for building a learning environment. In S. Granger, G. Gilquin and F. Meunier (eds.), *Twenty Years of Learner Corpus Research: Looking back, Moving ahead. Corpora and Language in Use-Proceedings 1*. Presses universitaires de Louvain, Louvain-la-Neuve: 493-503.

Wanner, L., S. Verlinde and M. Alonso Ramos 2013b. Writing assistants and automatic lexical error correction: word combinatorics. In Kosem, I., Kallas, J., Gantar, P., Krek, S., Langemets, M., Tuulik, M. (eds.), *Electronic lexicography in the 21st century: thinking outside the paper. Proceedings of the eLex 2013 conference, 17-19 October 2013, Tallinn, Estonia*. Institute for Applied Slovene Studies/Eesti Keele Instituut, Ljubljana/Tallinn: 472-487.

Wible, D., Nai-Lung Tsao. 2011. Towards a new generation of corpus-derived lexical resources for language learning. In F. Meunier, S. De Cock, G. Gilquin, M. Paquot (eds), *A taste for corpora. In honour of Sylviane Granger (Studies in corpus linguistics, 45)*. Benjamins, Amsterdam, Philadelphia: 237-254.

Wu, J.-C., Y.C. Chang, T. Mitamura and J. S. Chang 2010. Automatic Collocation Suggestion in Academic Writing. In *Proceedings of the ACL Conference*.

Turning garbage into a writing assistant

Serge Verlinde
KU Leuven, Leuven Language Institute / Dekenstraat 6, B-3000 Leuven
serge.verlinde@ilt.kuleuven.be

Abstract

In this paper, we discuss the development of two modules of a writing assistant for Dutch as a second or foreign language: a word combination checker and a module for error detection and correction based on the Google Web 1T 5-gram data set. The word combination checker differs from similar tools by its link with lexicographical data. The error detection and correction module is based on a simple n-gram approach.

1 Introduction

During the question and answer session after Adam Kilgarriff's talk at the Euralex congress in Oslo (Kilgarriff et al., 2012), Patrick Hanks referred to the use of web corpora for gathering linguistic data as *garbage in, garbage out*. And indeed, data from large web corpora often contain a lot of noise. However, for many research domains, such as lexicography (Kilgarriff, 2013), NLP (for an overview of corpora aimed at the NLP community, see http://www-nlp.stanford.edu/links/statnlp.html#Corpora) or error detection and correction (for an overview, see Leacock (2014)), (web) corpora are very helpful and are widely used by the research community. In this paper, we would like to present an ongoing project which uses the *Google Web 1T 5-gram, 10 European Languages Version 1* (Brants and Franz, 2009) to build a writing assistant for Dutch as a second or foreign language (*DS/FL*). Among other components, this writing assistant includes both a word combination dictionary and a proper error detection and correction module. A pilot version of these applications is already operational. The final version will be added to a writing assistant for (academic) Dutch, which is currently undergoing testing (De Wachter and D'Hertefelt, 2013; D'Hertefelt et al., 2014).

2 The Google Web 1T 5-gram data set

The Google Web 1T 5-gram data set offers n-grams and their observed frequency in a web corpus for 10 European languages: Czech, Dutch, French, German, Italian, Polish, Portuguese, Romanian, Spanish and Swedish.[1] As shown by the figures for Dutch in Table 1, this is a large-scale repository of data.

file sizes	2.8 GB compressed
Number of tokens:	133,771,492,564
Number of sentences:	16,751,987,759
Number of unigrams:	10,244,357
Number of bigrams:	65,334,723
Number of trigrams:	127,329,560
Number of fourgrams:	134,615,354
Number of fivegrams:	112,278,954
Number of n-grams:	449,802,948

Table 1: Google Web 1T 5-gram data set statistics for Dutch.

Google n-grams have recently been used to develop a variety of applications, such as spelling

[1] Details of the Google Web 1T data set can be found at http://catalog.ldc.upenn.edu/LDC2009T25. See also Evert (2010).

checkers (Bassil and Alwani, 2012) or error correction tools (Inkpen and Islam, 2011). The entire Google Web 1T 5-gram data set has also been made available for English by the *Corpus Linguistics group* at *FAU Erlangen-Nürnberg* (`http://corpora.linguistik.uni-erlangen.de/demos/cgi-bin/Web1T5/Web1T5_freq.perl`) and, using the same software package, by the *Information Science department* of the *University of Groningen* for Dutch (`http://www.let.rug.nl/gosse/bin/Web1T5_freq.perl`). The interface allows queries on frequency information, word associations and collocations. [2]

3 From data set to database

As the cut off frequency of the Google Web 1T 5-gram data set is 40, bigrams and trigrams may be lost in the fourgram and fivegram files. In order to preserve maximum data, we therefore decided only to retrieve bigram and trigram files.

To facilitate data handling, all lines containing non-relevant linguistic data were removed from the bigram and trigram files, as illustrated by following examples:

A 0 A 71
A 0 B 45
A 0 Het 155
A 0 Vraag 44
A 0 W 46

The original trigram files were compressed by 54%. In a next step, the reduced data set for both bigrams and trigrams was uploaded into a MySQL database. Separate tables were created for each letter. Lemmas and part of speech information were assigned to each of the bigrams and trigrams. Without context, all possible lemmas and parts of speech were linked to every word.

Finally, words were stored as integer values and clustered indexes were added to the relevant columns to make queries run faster. The current size of the database tables is 27 GB.[3] Structuring data in this way allowed us to optimize overall performances, although some queries still take some time to run.

[2]For more details, see Evert (2012)

[3]The current size of the English n-gram version is 211 GB for the whole data set (Evert, 2012).

4 The Leuven Language Institute writing assistants

Many tools may be considered as writing assistants: dictionaries, spelling and grammar checkers in word processors, online correction tools, collocation checkers, etc. Unfortunately for the user, these resources are only available separately. The writing assistants developed at the *Leuven Language Institute* try to combine these resources in order to facilitate the writing process. A first application was programmed for French as a second or foreign language. It is included in the *Interactive Language Toolbox*, an application offering access to the most relevant online lexicographical resources (*predictive writing aid*) as well as providing spell, grammar and lexical checking for French (*corrective writing aid*: Ziyuan (2012)). The Interactive Language Toolbox may be accessed at `http://ilt.kuleuven.be/inlato` (see also Verlinde and Peeters (2012)).

A second tool for (academic) Dutch is under construction (De Wachter and D'Hertefelt, 2013; D'Hertefelt et al., 2014). Like the French tool, it consists of a corrective writing aid with modules for spelling, style and text coherence and a predictive writing aid with search facilities for word definitions, academic alternatives for general language words, web examples and *Google Scholar* examples. The tool being developed for DS/FL will have the same interface, but with fewer modules. It will combine a spelling checker, an error detection and correction module and a predictive writing aid with, amongst others, a word combination checker. The error detection and correction module and the word combination checker are based on our reduced version of the Google Web 1T 5-gram data set for Dutch.

4.1 Word combination checker

Numerous word combination descriptions are available on the web for various languages, with the *SketchEngine* (`http://www.sketchengine.co.uk/`) being the most comprehensive tool.

A word combination checker is an interactive, online variant of these descriptions that suggests relevant words in a specific context

in answer to a users need. Some well-known examples for English are Netspeak (`http://www.netspeak.org`), Just the word (`http://www.just-the-word.com/`) and MUST (`http://miscollocation-richtrf.rhcloud.com/`), as well as the websites based on the Google Web 1T 5-gram data set referred to in Section 2.

The application that we programmed for Dutch is similar to these word combination checkers, but relies on enriched data: as explained above, we added part of speech information and we linked the bigrams and trigrams with lexicographical material. For instance, simplified semantic tags, inspired by Mel'čuk's lexical functions (Mel'čuk, 1996), were added to adjectives. We plan to tag more data, adverbs for instance, in the near future.

Two types of search functions, which reflect the actual needs of DS/FL users, are available:[4]

- search a word. An * in the query stands for any unknown or wildcard word in a specific context of maximum three words

  een * overwinning
  "a(n) * victory"
  ⟶ *belangrijke, grote, verdiende,*
  ...
  "important", "big", "deserved"

- search words with a specific part of speech. This query allows users to search for word combinations with a specific part of speech: which verb can I use with the noun *victory*? Which adjective meaning *big* can I use with the noun *victory*? What is the proposition used with the adjective *responsible*?

  een overwinning + verb
  ⟶ *behalen, vieren, ...*
  "gain", "celebrate"

More advanced search functions are also possible through specific encoding of the data:

- word combination patterns. What are the complex prepositions having the [preposition] + noun + [preposition] pattern for *bevel* "order, command"?

  ⟶ *[op] bevel [van], [onder] bevel [van], ...*
  "on the orders of", "under the command of"

More refined searches will be possible as semantic tagging is expanded: how can we intensify the verb *run*? How do we express the idea of *a lot of* in combination with the noun *cows*? etc.

The results of the searches shown in the examples above have been filtered before display. In the case of the *adverb + verb* pattern for instance, we retrieved all verb forms occurring after a given adverb, but we only display those with an infinitive. This seems the best way to increase precision, although the recall rate is somewhat lower. Tests will have to be undertaken to evaluate the impact of such filters more thoroughly.

From a didactic point of view, working with authentic data may offer a significant benefit over more analytical presentations of word combinations, as in the SketchEngine: natural sequences of words are presented to the user, demonstrating for instance the actual use of determiners or the preference for a plural form in certain contexts.

4.2 Error detection and correction

Leacock et al. (2014) provide an extensive overview of techniques used for automated error detection (and correction) and discuss the results of a considerable number of studies dedicated to this research topic. Not surprisingly, most of these studies focus on English and some very language specific problems encountered by many non-natives: the use of articles, prepositions and word combinations. Results are not always convincing or comparable (Leacock et al., 2014). Very recently, Wanner et al. (2013) made some alternative suggestions for dealing with word combinations in Spanish and French, illustrating the idea advanced by Gamon et al. (2009) that different techniques should tackle different error types.

For the error detection and correction tool for DS/FL, we decided to take a straight-forward approach, using our Google Web 1T 5-gram database. The user's text is split into sequences of three successive words. Each of these sequences

[4] Search functions are all programmed in PHP.

is compared to trigrams available in the database. If there is a match, the pointer moves to the next word and repeats the procedure. If there is no match, a set of heuristic rules are applied:

- a first rule searches for trigrams with the same lemmas

  *een academisch context
  $\longrightarrow$ *een academische context*
  "an academic context"

  These matches are suggested as possible corrections. (see Figure 1)

- a second rule searches for trigrams with a different article as gender confusion is a frequent error among learners of Dutch (neuter $><$ masculine/feminine)

  *de eerste voorbeeld
  $\longrightarrow$ *het eerste voorbeeld*
  "the first example"

  These matches are suggested as possible corrections.

- if neither of these rules yields a match, a third one splits the three-word sequence into 2 two-word sequences which are then matched with the bigrams in the database. If there is no match, the text is displayed in red. If the relative frequency of the match is below a cut off value (p=0.0001), the text is displayed in a smaller red font. In both cases, no corrections are suggested as these are mostly not relevant at all. (Figure 1)

In order to increase both speed and precision, we did not consider words beginning with a capital letter (except the first word of the sentence), punctuation marks, digits and words denoting numbers, days of the week or months of the year. Word combinations with these words are indeed numerous and not all of them occur in the bigram and trigrams files. We also excluded hyphenated words because it was used as a word boundary in the original Google Web 1T 5-gram data set.

The error detection and correction tool is a low-tech n-gram based application. However, for the first few evaluations performed on authentic texts from Dutch language learners at various levels, we achieved an acceptable precision rate of $>$60% and a recall rate of $>$50%.[5] These figures are slightly inferior to those reported by Inkpen and Islam (2011:16) on Romanian texts using Google n-grams (average precision for three texts: 73.30%, with a recall rate of 68.02%), However, it may be misleading to compare these results because Inkpen and Islam (2011) did not use authentic learners texts.

5 Conclusion

The few efficiency studies that we conducted to test our writing aids for French and (academic) Dutch (Rymenams et al., 2012; D'Hertefelt et al., 2014) have reinforced our belief in systems that assist non-natives in writing texts, even though no compelling scientific evidence exists for this claim (Leacock et al., 2014).

Writing assistants should combine text enrichment and text correction modules. Text enrichment tools already exist, but could benefit from additional, lexicographical information, thus raising search efficiency. A closer study of search results should help us identify areas for improvement.

The error detection and correction tool is, as far as we know, the first one designed for DS/FL. The results are encouraging, but here, too, improvements are needed. Leacock et al. (2014) argue that

> Any robust grammatical error detection system will be a hybrid system, using simple rules for those error types that can be resolved easily and more complex machine learning methods for those that cannot. Such a system may even need to fall back on parsing, despite all of its inherent problems, as n-grams (sequences of tokens) frequencies will not be effective for errors that involve long distance relations.

The fact that parsers are not entirely reliable when applied to language learner texts is one of

[5] Corpus of 4500 words, single rater as gold standard, any kind of error.

Figure 1: Error detection and correction output.

the main inherent problems. However, as language learners may benefit from error detection and correction tools, research should focus more on developing systems able to scan authentic texts for all possible errors. But the question of how to optimize such systems largely remains open.

References

Youssef Bassil and Mohammad Alwani. 2012. Context-sensitive Spelling Correction Using Google Web 1T 5-Gram Information. *Computer and Information Science*, 5(3):37-48.

Thorsten Brants and Alex Franz. 2009. *Web 1T 5-gram, 10 European Languages Version 1*. Linguistic Data Consortium, Philadelphia, PA. `http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2009T25`

Margot D'Hertefelt, Lieve De Wachter, and Serge Verlinde. 2014. Writing Aid Dutch. Supporting students' writing skills by means of a string and pattern matching based web application. In: *Proceedings of the 6th International Conference on Computer Supported Education*. vol. 1:486-491.

Lieve De Wachter and Margot D'Hertefelt. 2013. Writing Aid Dutch. A digital writing tool for university and college students. Paper presented before the *European Association for Practitioner Research on Improving Learning in education and professional practice*. (Biel, 27-29.11.2013).

Stefan Evert. 2010. Google Web 1T 5-Grams Made Easy (but not for the computer). In: *Proceedings of the NAACL HLT 2010 Sixth Web as Corpus Workshop*. 32-40.

Stefan Evert. 2012. Web N-Grams as a Resource for Corpus Linguistics. Paper presented at the *Computational Linguistics Colloquium*, Rupert Karls Universität, Heidelberg. `http://www.cl.uni-heidelberg.de/colloquium/docs/evert_web_ngrams_heidelberg2012.slides.pdf`

Michael Gamon, Claudia Leacock, Chris Brockett, William B. Dolan, Jianfeng Gao, Dmitriy Belenko, and Alexandre Klementiev. 2009. Using statistical techniques and web search to correct ESL errors. *CALICO Journal*, 26(3):491-511.

Diana Inkpen and Aminul Islam. 2011. Error Correction for English and Romanian Texts. In: Militon Frentiu, Horia F. Pop, and Simona Motogna (Eds.). *Proceedings of the International Conference on Knowledge Engineering, Principles and Techniques (KEPT 2011)*. 13-29.

Adam Kilgarriff. 2013. Using corpora as data sources for dictionaries. In: Howard Jackson (Ed.). *The Bloomsbury Companion to Lexicography*. London: Bloomsbury. 77-96.

Adam Kilgarriff, Pavel Rychlý, Vojtěch Kovář and Vít Baisa. 2012. *Finding Multiwords of More Than Two Words*. In: Ruth Vatvedt Fjeld and Julie Matilde Torjusen. Proceedings of the 15th EURALEX International Congress. Department of Linguistics and Scandinavian Studies, University of Oslo. 693-700.

Claudia Leacock, Martin Chodorow, Michael Gamon, and Joel Tetreault. 2014. *Automated Grammatical Error Detection for Language Learners, Second Edition*. Morgan & Claypool.

Igor A. Mel'čuk. 1996. Lexical Functions: A tool for the description of lexical relations in the lexicon. In: Leo Wanner (Ed.). *Lexical Functions in Lexicography and Natural Language Processing*. Amsterdam: John Benjamins. 37-102.

Sara Rymenams, Serge Verlinde, Steven Marx, Mieke Rosselle, and Laetitia Gerard. 2012. SOS français: Conception et évaluation d'un didacticiel d'aide à la rédaction interactif. In: *Proceedings of the Congrès Mondial de linguistique française (CMLF 2012)*. 377-393.

Serge Verlinde and Geert Peeters. 2012. Data access revisited: The Interactive Language Toolbox. In: Sylviane Granger and Magali Paquot (Eds.). *Electronic Lexicography*. Oxford: Oxford University Press. 147-162.

Leo Wanner, Serge Verlinde, and Margarita Alonso Ramos. 2013. Writing assistants and automatic lexical error correction: word combinatorics. In: Iztok Kosem, Jelena Kallas, Polona Gantar, Simon Krek, Margit Langemets, Maria Tuulik (Eds.). *Electronic lexicography in the 21st century: thinking outside the paper*. Proceedings of the eLex

2013 conference, 17-19 October 2013, Tallinn, Estonia. Ljubljana/Tallinn: Trojina, Institute for Applied Slovene Studies/Eesti Keele Instituut. 472-487. http://eki.ee/elex2013/proceedings/eLex2013_33_Wanner+Verlinde+Alonso-Ramos.pdf

Yao Ziyuan. 2012. *Breaking the Language Barrier: A Game-Changing Approach*. E-book available at https://sites.google.com/site/yaoziyuan/publications/books/breaking-the-language-barrier-a-game-changing-approach

A lexical database for systematic orthographical teaching and training of German orthography

Hrovje Hlebec
University Hildesheim
Institute for
German Language
and Literature

Wilfried Hehr
University Hildesheim
Institute for
Economics and
Business Informatics

Ronny Jauch
University Hildesheim
Institute for
Information Science and
Natural Language Processing

{hlebec,hehrwi,jauchr}@uni-hildesheim.de

Abstract

This paper introduces a tool which is still under development, consisting of a lexical database and suitable query interfaces, for supporting systematic orthographic instruction. The first part of the paper is an introduction to the conceptual base for the project. The second part describes the technical implementation in several steps: It first presents the user profile and the underlying database structure before explaining an algorithm which we used to make some of the database contents more explicit.

1 Objective

The purpose of the system is to enable teachers to access German word material in a structured manner for systematic orthographic instruction; in our case – contrary to conventional thinking – orthographic instruction is regarded not only as writing instruction, but also as reading instruction (Noack, 2010, cf.). Therefore, the structure and function of orthographic regularities will be described below mainly with respect to the reading process.

2 Conceptual Base

The basis for systematic orthographic instruction is the scientific modeling of written language structures, which regards written language as based in spoken language, but does not reduce the relationship between the two to a mapping – a perspective that has been particular to linguistics for a long time and has likely been applied often in instruction in the past. The orientation towards the writing system goes hand in hand with the focus on the core area of the lexicon. The core area selected here is the set of words whose structure follows the central regularities of the writing system. Working on prototypical word material should enable learners to acquire these regularities with the least number of errors, both in explicit and implicit learning processes.

2.1 Orthography theoretical Background

In the modeling of the core area, we essentially follow Eisenberg's concept of the core word (Eisenberg, 2011, p. 18ff.). In order to be entered into the database as a core word, a lexeme must meet the following criteria: It is a simplex whose paradigm exhibits at least one disyllabic form with its stucture consisting of a stressed main syllable and an unstressed reduced syllable, i.e. a trochaic foot. Thus, this excludes words such as *Geflügel* and *Hühnchen*, which are morphologically complex and also derivable based on the basic regularities, *Salat* and *Kamel*, which exhibit iambic foot, and *Papi* and *Iglu*, which are in fact trochaic, but still end in full vowels. The database currently contains nearly 3,300 lexemes that meet these requirements. Database access is handled using an interface whose structure is geared towards what is known as the "house/garage" model (hereinafter the "HG model"), (Bredel, 2009, cf.).

This model visualizes both the basic trochaic foot (with main syllable and reduced syllable) and the internal syllable structure with the constituents of onset, nucleus and coda, (abbreviated, in Fig. 1 as O, N, K, respectively), to enable struc-

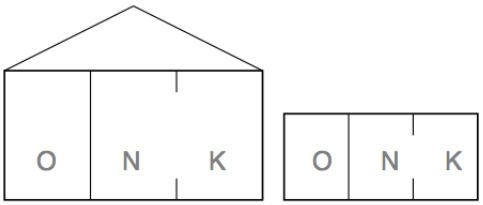


Figure 1: The house/garage modell (HG modell)

tured access to orthographic patterns for learners. Unlike traditional designs, which assign a phonetic value to isolated letters, the HG model can help illustrate the fact that letters have a phonetic potential, the actual realization of which depends on the position and distribution of the letters in the word. This can be demonstrated succinctly with the letter <e> in German: "While the absolute position of <e> determines whether <e> must be recoded as a full vowel (main syllable articulation) or a reduced vowel (reduced syllable articulation), the relative position, i.e. the distribution of <e> within the syllable, determines the precise vowel quality that must be selected." (Bredel, 2009, p. 139, our translation). If the coda of the main syllable is occupied, then <e> must be recoded as a lax short vowel; if it is unoccupied, then <e> is recoded as a tense long vowel (see Fig. 2). Analogous regularities are found in the reduced syllable (Bredel, 2009, cf. p. 139) .

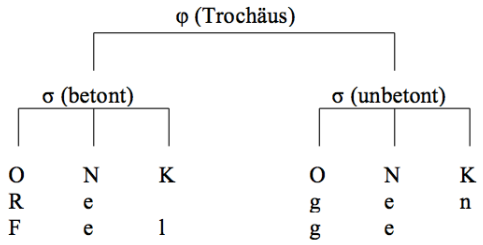


Figure 2: base types

2.2 Queryable phenomena

The database enables users to search by individual syllable positions in the main and reduced syllables (Kasse, nennen). Only the nucleus of the reduced syllable, which is occupied by <e> in all core words, is excluded here.
For syllable constituents that can contain consonant letters, queries are possible both by the number of letters and by the letters themselves (including letter combinations). In such queries, the letters can be entered freely. For the nucleus of the main syllable, the user can select from eligible vowel letters and orthographic diphthongs.
Queries can be made more specific using an additional menu:

1. The "Orthographic Regularities" menu item can be used to retrieve – in a targeted manner – words that exhibit written indicators of shortness or length. These written indicators include syllable-joint-spelling (Silbengelenkschreibung), syllable initial <h> and what is known as the "Dehnungs-h" (Eisenberg, 2013, pp. 299).
With syllable-joint-spellings, the spoken form features an ambisyllabic consonant, i.e. an internuclear consonant belonging to both the main and reduced syllables. In writing, the corresponding consonant letter is doubled, resulting in a written word form in which the coda of the main syllable is occupied (see Fig. 3). Syllable-joint-spellings also include the written forms <ck> and <tz> as in *Deckel* and *Stütze*.

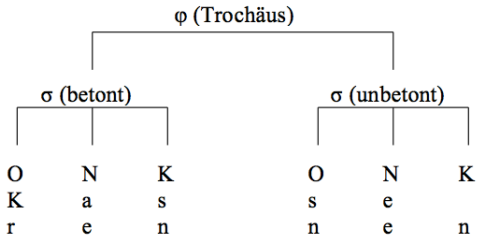


Figure 3: Syllable-joint-spelling

Syllable initial <h> has no phonemic expression at the segmental level. It occurs when a stressed open syllable and an unstressed naked syllable follow one another. In writing, syllable-initial <h> occupies the onset of the reduced syllable, making the syllable structure visually salient (nahen, Ruhe, see Fig. 4).

In contrast, the Dehnungs-h occupies the coda of the main syllable (see Fig. 5). This only occurs when the onset of the reduced syllable is occupied by the consonant letters <l, r, m, n>. However, this structure is

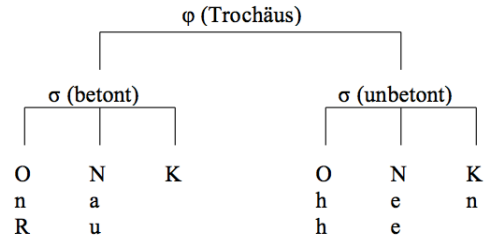


Figure 4: Syllable initial <h>

found (lehnen, Fehler) only in about half of the possible cases. According to Eisenberg, its function consists first and foremost in indicating the tense articulation of the main syllable's vowel (Eisenberg, 2013, cf. p. 303).

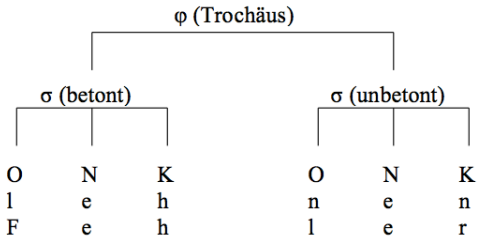


Figure 5: Dehnungs-h ("lengthening h")

2. The "Grammatical Form" menu item can be used to specify whether the system should return only words whose lemma is disyllabic, or also words whose disyllabic form deviates from the lemma. While the lemma is always disyllabic for verbs, this is largely not the case with adjectives. Therefore, for adjectives, a form in the nominative masculine singular always appears in the database e.g. *rote*, *schnelle*. Around two thirds of the noun lemmas are disyllabic. The remaining third are given either in the nominative plural (*Spieße*, *Tänze*) or – if no plural form exists – the genitive singular (*Rapses*, *Sandes*).

3. The "Individual Phenomena" menu item can be used to retrieve a series of words that appear to be perfectly regular, but which cannot be represented properly under the HG model. These are words that have internuclear <-ch-> or <-sch-> letter sequences in writing and an ambisyllabic consonant in speech. While in standard cases, the consonant letter is doubled (see above), this is not the case in polygraphs (**Taschsche*, **Küchche*) (Eisenberg, 2013, cf. p. 300).

4. The database also enables the specification of the part of speech in search queries. From the perspective of orthographic theory, the part of speech of the expressions admittedly does not play a role. However, for instructional purposes, it may in fact be advantageous to have access to a word inventory sorted by part of speech if orthographically relevant morphological phenomena need to be examined according to their part of speech, based on the basic regularities. Thus, for instance, it is possible to use nouns with monosyllabic singulars and disyllabic plurals specifically, in order to work on the phenomenon of stem constancy: For example, while [zi:bə] has a voiced plosive in the onset of the reduced syllable, it is devoiced in the monosyllabic form due to phonological regularities (terminal devoicing: [zi:p]), (Wiese, 2000, for terminal devoicing in German, cf. pp. 200). In writing, the stem is written identically in all forms wherever possible, so it remains easily identifiable to the reader across all morphological contexts (*Sieb*, *Siebe*, *Siebchen*, *Siebdruck*).

Although, on the whole, morphology-based orthographic regularities admittedly play a subordinate role in the database design, it should also be noted here that it is possible to work on morphology-based orthographies under the HG model (Wiese, 2000, for terminal devoicing in German, cf. pp. 200). The search interface takes this into account by means of colored highlighting at the stem boundaries, standardly located between the onset and the nucleus of the reduced syllable.

2.3 Possible Uses and Output

The main application of the database for teachers consists, on the one hand, in providing access to a word inventory that meets all requirements for systematic orthographic instruction. On the other hand, they also have the option to perform targeted searches for word material in order to deal

with specific orthographic phenomena, and to use these in their instruction.

To facilitate its application in education, the database offers various output options:

First off, teachers can have the system output a word list, prepared by the teacher, as a simple text document: so the words they would like to use can be integrated seamlessly into their own instructional materials. In addition to this, teachers also have the option to use two worksheet templates. Both templates offer the option to enter a title, a specific work assignment and additional instructions in the text fields provided to this effect. The first worksheet template outputs the word material in table form so the teachers can make minor graphical adjustments. The second worksheet template presents the word material such that it can be used with the "Leselineal" ("reading ruler") – a tool currently under development for reading instruction based on the system outlined above.[1]

In addition to this, the database homepage also provides teachers with a house/garage template that can be used to create teaching aids independently. When using the HG model, teachers will receive additional support because the database lets them select the "Arrange" menu item for each word or word list to show its arrangement in the HG model.

3 Technical Architecture and User Interaction

This system is intended for teachers looking for material for systematic orthography courses. Learners are only considered as users in the second line. So no exercises are offered, but the word material classified according to the principles discussed in chapter 2 can be used to create samples and exercises.

3.1 Introduction

Figure 6 visualizes the elements of the architecture: The user goes to a central web site and defines certain selection criteria to receive matching results.

The design of the web project is realized using easily modifiable and thereby future-proof `CSS3`

[1]The tool is being developed by Melanie Bangel, Ursula Bredel, Gabriele Hinney, Astrid Müller & Tilo Reisig.

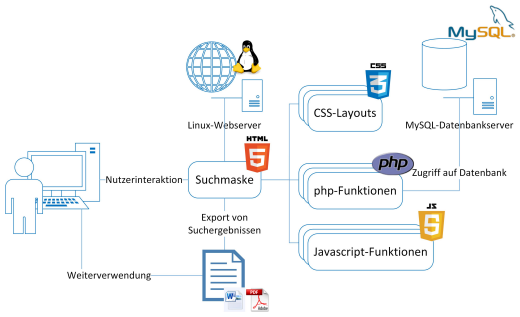


Figure 6: Technical concept

layouts in combination with `HTML5` code. This design is embedded with results of `PHP` functions (e.g. results of database inquiries), which are organized in external files to ensure clarity and modularity. Crucial to the project are the maintenance and the extension of a lexical database implemented in `MySQL`. The dynamic programming language `JavaScript` is used to make this interaction possible and to ensure an intuitive handling and good usability. The `JavaScript` methods are also filed externally. We are currently (autumn 2014) working on providing an export function for search results, e.g. to create work sheets. Appropriate formats can be **.txt*, **.doc* and **.pdf*.

3.2 Explaining the current database scheme

Figure 7 shows a diagram of the current scheme of the database: Central is a list of about 3.300 German words as well as different binary features recorded for these words.

Table 1 shows the terms used in the database instead of those used in section 2 and in linguistic theory; we provide these "aliases" from didactic grammar, as the target group might not be familiar with the terms introduced in chapter 2.

Term	**Term in database**
Onset der HS	*Anfangsrand 1* (*AR1*)
Nucleus der HS	*Kern 1* (*K*)
Coda der HS	*Endrand 1* (*ER1*)
Onset der RS	*Anfangsrand 2* (*AR2*)
Coda der RS	*Endrand 2* (*ER2*)

Table 1: Terms in theory and in the database

At present the columns cover word form, word class (=POS) (WA), initial margin 1 (AR1, =on-

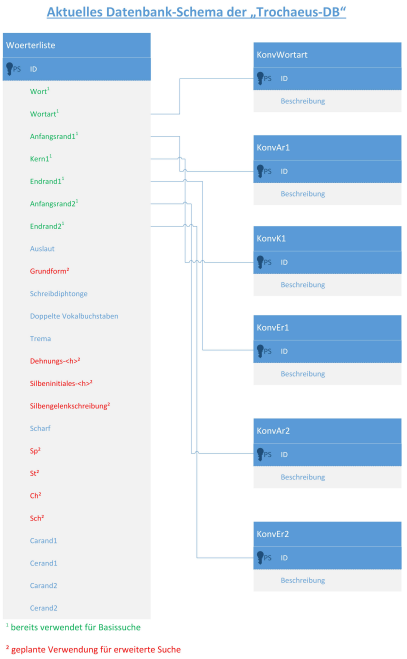


Figure 7: Database scheme

set), nucleus 1 (K), final margin 1 (ER1, = coda), initial margin 2 (AR 2) and final margin 2 (ER2). Table 2 shows several examples:

ID	Wortform	WA	AR_1	K	ER_1
1	aber	4	0	a	0
3	Achse	1	0	a	3
290	decken	2	1	e	1
2709	tapfer	3	1	a	1

Table 2: Representation of word characteristics in the database

ID	Wortform	AR_2	ER_2
1	aber	1	1
3	Achse	2	0
290	decken	1	1
2706	tapfer	1	1

Table 3: Table 2 continued

Each entry has a unique ID. The words are classified as nouns (1), verbs (2), adjectives (3) or others (4). The numbers in columns AR1, ER1 and AR2 stand for the number of letters which occupy the respective positions. In column K then, the actual letters that make up the nucleus of the first syllable are given (see chapter 2.2). As it is inconvenient for the user to work with IDs instead of actual letters, charts converting each ID coded criterion are given. These charts show the meaning of each ID, which is then used to work with in the search screen. Table 4 shows the meaning of the interface data for the choice of the characteristics of the onsets of the main syllables: Thanks to such charts (remaining tastes), the meta language used within the interface can easily be adjusted to the needs of users of different skill levels; a choice could be offered of e.g. scientific terms or terms used in different types of teaching or learning material. This is an element of individualisation and user adaptivity.

ID	Beschreibung
0	nicht belegt
1	1 Buchstabe
2	2 Buchstaben
3	3 Buchstaben
4	4 Buchstaben

Table 4: Convention table AR1

3.3 Choice of relevant selection criteria using a search screen

Figure 8 shows a screenshot of the currently implemented search screen for the selection of the criteria defined above. By its graphical design the search interface supports the HG-modell. In the list ”available” the user is shown available options for each criterion.

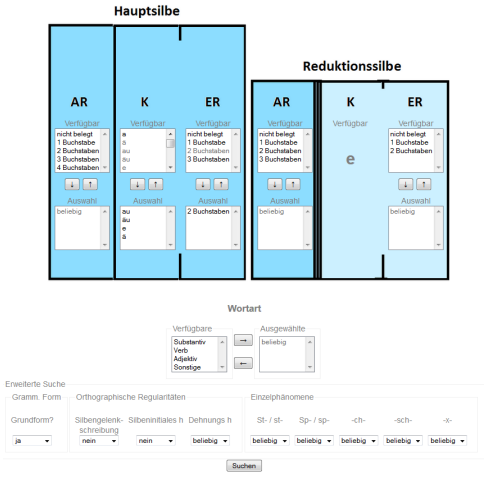


Figure 8: Search interface

These can then be chosen by double click or by the use of the buttons in the list labelled

”Auswahl” (”choice”). The chosen entry in the list ”verfügbar” (”available”) is then disabled. In the same way chosen options can be removed by double click or by using the buttons. At the first opening of the search screen, prior to any selections of the user, the choice list states ”beliebig” (”any”). This means that no special options of a selection criterion have been chosen so that in each case all alternative values of the respective criterion are possible results. Then word class values can be chosen so that only words of the chosen part of speech (along with their chosen characteristics) will be shown. After the search is sent, the results are presented underneath the search zone.

3.4 Presentation of previously defined selection criteria

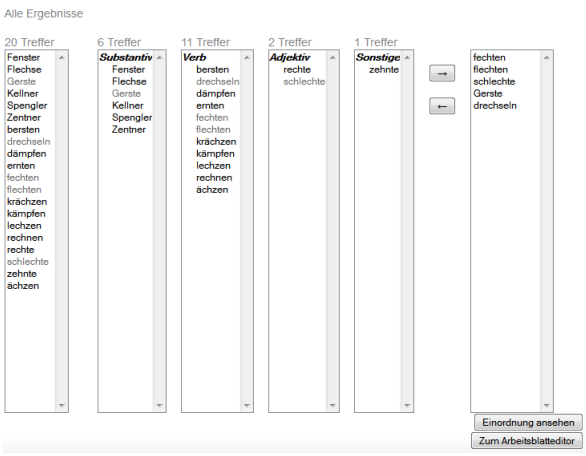


Figure 9: Presentation of results

Figure 9 shows the presentation of matching results. On the one hand all the results are presented in alphabetical order (left column). On the other hand, in a parallel placement, there are separate lists of results for each word class. For further processing of the results, e.g. for the use of the output options, there is a list of selected items. It can be filled by the user with specially selected words from the result lists. This can be done either by double click on any word given in one of the five result lists or by using the buttons. Selected words are disabled in the result lists for convenience.

4 Database extension

The structure of this database contains the following six columns *Wortform*, *AR1*, *K*, *ER1*, *AR2* and *ER2* (cf. Table 1). Each of the columns *AR1*, *ER1*, *AR2* and *ER2* contains the number of letters (see above section 3) but the graphemical image is not yet part of the database structure. The column *Wortform* contains the whole word and the column *K* contains the graphemical image of the main syllable nucleus). Without the full set of graphemical images, one may not create queries like ”column *AR1* contains <sch>”. In order to get this result with the given information, one would have to create a query like ”Compare the number of column *AR1* with 3 and check, if the column *wordform* contains <sch> and the word starts with the same <sch>.”. This kind of query does not have a good performance, which is why this approach is not in the focus anymore and we replaced it with an offline extension of the database. In order to accomplish this task, we implemented a script in the dynamic programming language `Python`. The script generates the graphemical images on the basis of the numbers in the columns *AR1*, *ER1*, *AR2* and *ER2* and the information of the columns *wortform* and *K* and writes the output into a file in `csv`-format (*comma separated values*).

wortform	AR1	K	ER1	AR2	ER2
Bäume	1	äu	0	1	0
Sträucher	3	äu	0	2	1
Schläuche	4	äu	0	2	0
Kräuter	2	äu	0	1	1
löschen	1	ö	99	99	1
mischen	1	i	99	99	1

Figure 10: State before processing the algorithm[2]

The following paragraph explains the algorithm which computes the graphemical images. The algorithm runs through every line and it memorizes the number of letters in the columns *AR1*, *K*, *ER1*, *AR2* and *ER2*: the algorithm needs those values in order to compute the corresponding graphemic images. For example, to compute the corresponding graphemic image of *ER1*, you

[2]The number ”99” is an exception marker which indicates that the wordform can not be represented in terms of the HG-modell.

need to know the left and the right border of *ER1*. The left border is given by the sum of *AR1* and *K* and the right border is given by the sum of *AR1*, *K* and *ER1*. Such a calculation is done for every syllabe field. Exceptions are the column *ER1* and *AR2*, if their corresponding wordform contains an exception marker (”99”). Wordforms with the parts *-ch-*, *-sch-*, *-x-* are marked with an exception marker. Those may not be mapped, because a mapping is not unambigously possible in terms of orthography theory (cf. above section 2.2).
The result of this procedure is a *csv*-file which contains the following additional columns: graphemic start of main syllable (*GAR1*), graphemic end of main syllable (*GER1*), graphemic start of reduction syllable (*GAR2*) and graphemic end of reduction syllable (*GER2*). This file is imported into the existing database. Figure 10 shows the state before processing the graphemic columns and figure 11 shows a section of the state after that process.

wortform	AR1	K	ER1	AR2	ER2	GAR1	GER1	GAR2	GER2
Bäume	1	äu	0	1	0	B		m	
Sträucher	3	äu	0	2	1	Str		ch	r
Schläuche	4	äu	0	2	0	Schl		ch	
Kräuter	2	äu	0	1	1	Kr		t	r
löschen	1	ö	99	99	1	l			n
mischen	1	i	99	99	1	m			n

Figure 11: State after processing the algorithm

Since the project database was created manually and is supposed to be updated manually, it is necessary to be able to identify potential annotation errors. Hence, the script has an error logging system which prints out every line on which the length of the wordform is not equal to the sum of the columns (*AR1*, *K*, *ER1*, *AR2* and *ER2*), for example. After the script is done, there is a possibility to save all errors in a separate file. With the help of this file, one may manually correct the data source in order to get a high quality `csv`-file, when the data correction is completed, one may import the result into the database.

5 Summary

At this time, the database project features a basic inventory of data (nearly 3,300 entries) and an initial interface version. The amount of entries available will continuously be increased, though we cannot currently predict with accuracy how many lexemes will correspond to the underlying core word definition. According to Eisenberg (2013), German features approx. 10,000 morphologically simple, independent words, so this should be the maximum number of projected entries. The basic inventory has already been migrated to a MySQL database and connected with the interface. At this time, some features are queryable in the interface (word form, part of speech, onset 1, nucleus 1, coda 1, onset 2, nucleus 2, coda 2). In addition, the database has also been expanded with graphemic substrings by means of an offline expansion.

6 Further steps

One major task for the future consists in drafting explanatory notes on the conceptual framework for orthographic instruction and, building on this, a tutorial for using the search interface.
To meet teachers’ needs to the greatest extent possible, additional phenomena relevant to language training could be made accessible through specific queries, such as words that have <st> or <sp> in the onset of their main syllables, or words that exhibit forms in speech that are subject to the regularities of terminal devoicing. Some of the data needed for the implementation of such additions are already available.
According to current planning, administrator and user roles will be separated. Administrators will be able to edit the database using the interface. Standard users will be able to create an account where they can save the word lists and worksheets they have created and access them at any time. A feedback function will enable all users to submit suggestions or proposals for improvements to the administrator. Another possible function could be offering users contexts of the word forms by means of corpus extracts.

References

Ursula Bredel. 2009. Orthographie als System - Orthographieerwerb als Systemerwerb. *Zeitschrift für Literaturwissenschaft und Linguistik*, 153:135 – 155.

Peter Eisenberg. 2011. *Das Fremdwort im Deutschen*. De Gruyter, Berlin/New York.

Peter Eisenberg. 2013. *Grundriss der deutschen Grammatik*, volume 1: Das Wort. Unter Mitarbeit

von Nanna Fuhrhop. Metzler, Stuttgart/Weimar, 4. aktualisierte und überarbeitete Auflage edition.

Christina Noack. 2010. Orthographie als Leserinstruktion. Die Leistung schriftsprachlicher Strukturen für den Dekodierprozess. In Ursula Bredel, editor, *Schriftsystem und Schrifterwerb: linguistisch - didaktisch - empirisch*, pages 151 – 170. De Gruyter, Berlin/New York.

Richard Wiese. 2000. *The Phonology of German*. Oxford University Press, Oxford.

GermEval

GermEval 2014 Named Entity Recognition Shared Task: Companion Paper*

Darina Benikova*, Chris Biemann*, Max Kisselew†, Sebastian Padó†
* Language Technology, TU Darmstadt, Germany
† Institute for Natural Language Processing, Universität Stuttgart, Germany
{darina.benikova@stud,biem@cs}.tu-darmstadt.de
{max.kisselew,pado}@ims.uni-stuttgart.de

Abstract

This paper describes the GermEval 2014 Named Entity Recognition (NER) Shared Task workshop at KONVENS. It provides background information on the motivation of this task, the data-set, the evaluation method, and an overview of the participating systems, followed by a discussion of their results. In contrast to previous NER tasks, the GermEval 2014 edition uses an extended tagset to account for derivatives of names and tokens that contain name parts. Further, nested named entities had to be predicted, i.e. names that contain other names. The eleven participating teams employed a wide range of techniques in their systems. The most successful systems used state-of-the-art machine learning methods, combined with some knowledge-based features in hybrid systems.

1 Introduction

Named Entity Recognition (NER or NERC) is the identification and classification of proper names in running text. NER is used in information extraction, question answering, automatic translation, data mining, speech processing and biomedical science (Jurafsky and Martin, 2000).

The starting point for this shared task is the observation that the level of performance of NER for German is still considerably below the level for English although German is a well-researched language. At least part of the reason is that in English, capitalization is an important feature in detecting Named Entities (NEs). In contrast, German capitalizes not only proper names, but all nouns, which makes the capitalization feature much less informative. At the same time, adjectives derived from NEs, which arguably count as NEs themselves, such as *englisch* ("English"), are not capitalized in German, in line with "normal" adjectives. Finally, a challenge in German is compounding, which allows to concatenate named entities and common nouns into single-token compounds.

This paper reports on a shared task on Named Enitity Recognition (NER) for German held in conjunction with KONVENS 2014. Compared to the only well-known earlier shared task for German NER held more than ten years ago in the context of CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003), our shared task corpus introduces two substantial extensions:

Fine-grained labels indicating NER subtypes. German morphology is comparatively productive (at least when compared to English). There is a considerable amount of word formation through both overt (non-zero) derivation and compounding, in particular for nouns. This gives rise to morphologically complex words that are not identical to, but stand in a direct relation to, Named Entities. The Shared Task corpus treats these as NE instances but marks them as special subtypes by introducing two fine-grained labels: `-deriv` marks derivations from NEs such as the previously mentioned *englisch* ("English"), and `-part` marks compounds including a NE as a subsequence

deutschlandweit (“Germany-wide”).

Embedded markables. Almost all extant corpora with Named Entity annotation assume that NE annotation is “flat”, that is, each word in the text can form part of at most one NE chunk. Clearly, this is an oversimplification. Consider the noun phase *Technische Universität Darmstadt* (“Technical University (of) Darmstadt”). It denotes an organization (label `ORG`), but also holds another NE, *Darmstadt*, which is a location (label `LOC`). To account for such cases, the Shared Task corpus is annotated with two levels of Named Entities. It captures at least one level of smaller NEs being embedded in larger NEs.

In summary, we distinguish between 12 classes of NEs: four main classes `PER`son, `LOC`ation, `ORG`anisation, and `OTH`er and their subclasses, annotated at two levels (“inner” and “outer” chunks). The challenge of this setup is that while it technically still allows a simple classification approach it introduces a recursive structure that calls for the application of more general machine learning or other automatically classifying methods that go beyond plain sequence tagging.

2 Dataset

The data used for the GermEval 2014 NER Shared Task builds on the dataset annotated by (Benikova et al., 2014)[1]. In this dataset, sentences taken from German Wikipedia articles and online news were used as a collection of citations, then annotated according to extended NoSta-D guidelines and eventually distributed under the CC-BY license[2].

As already described above, those guidelines use four main categories with sub-structure and nesting. The dataset is distributed contains overall more than 31,000 sentences with over 590,000 tokens. Those were divided in the following way: the training set consists of 24,000 sentences, the development set of 2,200 sentences and the test set of 5,100 sentences. The test set labels were not available to the participants until after the deadline. The distribution of the categories over the whole dataset is shown in Table 1. Care was taken to ensure the even dispersion of the categories in the subsets.

Class	All	Nested[3]
Location	12,204	1,454
Location deriv	4,412	808
Location part	713	39
Person	10,517	488
Person deriv	95	20
Person part	275	29
Organization	7,182	281
Organization deriv	56	4
Organization part	1,077	9
Other	4,047	57
Other deriv	294	3
Other part	252	2
Total	41,124	3,194

Table 1: Distribution of classes in the entire dataset of 31,300 sentences. Counts differ slightly fron what was reported in (Benikova et al., 2014) due to correction of inconsistencies in June 2014.

The entire dataset contains over 41,000 NEs, about 7.8% of them embedded in other NEs (*nested* NEs), about 11.8% are derivations (*deriv*) and about 5.6% are parts of NEs concatenated with other words (*part*).

The tab-separated format used in this dataset is similar to the CoNLL-Format. As illustrated in Table 2, the format used in the dataset additionally contains token numbers per sentence in the first column and a comment line indicating source and data before each sentence. The second column contains the tokens. The third column encodes the outer NE spans, the fourth column the inner ones. The BIO-scheme was used in order to encode the NE spans. In our challenge, further nested columns were not considered.

3 Evaluation method

We defined four metrics for the shared task, but only one was used for the final evaluation (“official metric”). The others were used in order to gain more insight into the distinctions between the

[1]The dataset was updated for this task to fix some inconsistencies.

[2]This license allows to distribute, alter and mix the data in any possible way and to use it for any purpose, including commercial ones (see `https://creativecommons.org/licenses/by/3.0/de/`).

[3]These numbers include all occurrences on the second level, regardless of the class of the first level NE

#	http://de.wikipedia.org/wiki/Manfred_Korfmann		
1	Aufgrund	O	O
2	seiner	O	O
3	Initiative	O	O
4	fand	O	O
5	2001/2002	O	O
6	in	O	O
7	Stuttgart	B-LOC	O
8	,	O	O
9	Braunschweig	B-LOC	O
10	und	O	O
11	Bonn	B-LOC	O
12	eine	O	O
13	große	O	O
14	und	O	O
15	publizistisch	O	O
16	vielbeachtete	O	O
17	Troia-Ausstellung	B-LOCpart	O
18	statt	O	O
19	,	O	O
20	,,	O	O
21	Troia	B-OTH	B-LOC
22	-	I-OTH	O
23	Traum	I-OTH	O
24	und	I-OTH	O
25	Wirklichkeit	I-OTH	O
26	”	O	O
27	.	O	O

Table 2: Data format illustration. The example sentence contains five named entities: the locations “Stuttgart”, “Braunschweig” and “Bonn”, the noun including a location part “Troia”-Ausstellung, and the title of the event, “Troia - Traum und Wirklichkeit”, which contains the embedded location “Troia”. (Benikova et al., 2014)

different systems.

We follow the pattern of previous evaluation in NER shared tasks using non-recursive data, which used the standard precision, recall and F_1 score metrics, using each individual markable as a datapoint in the P/R calculation. Let P denote the set of NE chunks predicted by a model and G the set of gold standard chunks. Precision, Recall, and F_1 are usually computed on the basis of of true positives and false positives and negatives, defined by set theoretic operations, e.g. $TP = P \cap G$ which in turn build on the definition of matches between predicted chunks and gold standard chunks. Normally, strict match is assumed: $p == g$ iff $label(p) = label(g)$ and $span(p) = span(g)$.

We would like to retain precision and recall as evaluation measures but need to redefine their computation to account for the nested nature of the data. Let P_1 and G_1 denote the set of all “first-level”/“outer” NEs (and P_2 and G_2 denote the set of all “second-level”/“inner” NEs in the predictions and in the gold standard, respectively.

3.1 Metric 1: Strict, Combined Evaluation (Official Metric)

The most straightforward evaluation treats first-level and second-level NEs individually and independently. This can be modeled by combining G and P across levels, but taking the level into account in the match definition:

$$\begin{aligned} P &= P_1 \cup P_2 \\ G &= G_1 \cup G_2 \\ p == g \text{ iff } &label(p) = label(g) \text{ and} \\ &span(p) = span(g) \text{ and} \\ &level(p) = level(g) \end{aligned}$$

Thus, this metric distinguishes all 12 labels (4 NE types, each in base, deriv and part varieties) and treats all markables on a par. It is used to determine the overall ranking of the systems in this challenge.

3.2 Metric 2: Loose, Combined Evaluation

Metric 2 again treats each NE individually but we collapse the label subtypes (base, deriv, part) so that a match on the base NE class is sufficient. For example, PER matches PERderiv:

$$\begin{aligned} P &= P_1 \cup P_2 \\ G &= G_1 \cup G_2 \\ p == g \text{ iff } &baseLabel(p) = baseLabel(g) \text{ and} \\ &span(p) = span(g) \text{ and} \\ &level(p) = level(g) \end{aligned}$$

This metric is useful to quantify the quality of systems at a coarse-grained level. It also makes the scores better comparable to previous NER evaluations, which have mostly used only four labels.

3.3 Metric 3: Strict, Separate Evaluation

Finally, this evaluation computes two sets of P/R/F1 values, one for G_1/P_1 and one for G_2/P_2. This metric considers the first-level and second-level markables separately which allows us to see

System ID	Institution
Nessy	LMU Munich
NERU	LMU Munich
HATNER	LMU Munich
DRIM	LMU Munich
ExB	ExB GmbH
BECREATIVE	LMU Munich
PLsNER	TU Darmstadt
mXS	University of Tours
MoSTNER	Marmara University
Earlytracks	EarlyTracks S.A.
UKP	TU Darmstadt

Table 3: Participants of the GermEval 2014 shared task.

how well systems do on first-level vs. second-level markables individually. It uses strict matching of labels, and thus uses exactly the traditional match definition (cf. the beginning of Section 3).

4 Participating systems

11 teams listed in Table 3 participated in the GermEval 2014 challenge. In the first subsection their general approaches will be discussed. The second subsection will present the variety of features that was used by the systems. Although many teams experimented with other methods and features, only those used by the respective final system will be mentioned here.

4.1 Methods used by the participants

Table 4 shows the different approaches the teams used for their NER systems. The first two columns describe handcrafted rules or gazetteer queries as an individual processing step, when not used merely as a feature in the overall system.

The NERU (Weber and Pötzl, 2014) system uses handcrafted rules made individually for the classes PERson, LOCation and ORGanization. Hence it is the only participating system not using any machine learning (ML).

The table shows that four systems (Nessy (Hermann et al., 2014), HATNER (Bobkova et al., 2014), EarlyTracks (Watrin et al., 2014), and BECREATIVE (Dreer et al., 2014)) use a hybrid approach, combining a ML method with handcrafted rules or gazetteer queries. All three systems use ML in the first step of their classification and some sort of gazetteer look-up as a post-processing step. Both Nessy and BECREATIVE use NB in the first step of their system, whereas HATNER uses ME. Nessy and HATNER do so only for the part and deriv classification using handcrafted rules.

System	HR	GQ	NB	ME	SVM	CRF	NN
NERU	X						
Nessy	X		X				
HATNER	X			X			
DRIM					X		
EarlyTracks	X	X				X	
ExB				X[4]		X	
BECREATIVE		X	X				
PLsNER							X
mXS				X			
MoSTNER						X	
UKP							X

Table 4: Methods used by participating systems HR = handcrafted rules, GQ = gazetteer queries, NB = Naïve Bayes, ME = Maximum Entropy, SVM = Support Vector Machine, CRF = Conditional Random Field and NN = Neural Networks/Word Embeddings

The goal of the ExB group (Hänig et al., 2014) was to build a system that runs efficiently on mobile devices. They experimented with different ML mechanisms. The result of their experiment was that the system that found more correct NEs made use of CRFs, but recommend to use ME in situations where resources are limited.

All other groups decided for one ML mechanism only. DRIM (Capsamun et al., 2014) uses SVM, ExB Group, and MoSTNER (Schüller, 2014) use CRF, and PLsNER (Nam, 2014) and UKP (Reimers et al., 2014) use NN.

4.2 Features used by the participating systems

Table 5 displays the types of features used by the participating systems. As NERU used gazetteers for its handwritten rules, it made no use of any other features. As shown, all systems except PLsNER made use of gazetteers and POS-tags.

5 Discussion of results

This section provides and discusses the results of the submitted systems.

5.1 Analysis by official metric (M1)

Table 6 shows the results of the systems in terms of M1, the official metric. For the sake of clarity, we

[4]More efficient, but lower prediction quality than CRF

System	G	POS	tok	NE-n	cap	NE	lem	1st	last	tok-n	#span	POS-n	char	WS	KW	SeC	SiC	WE
NERU	X																	
Nessy	X	X	X	X	X	X	X	X	X	X	X							
HATNER	X	X	X				X			X		X						
DRIM	X	X	X				X			X			X	X	X			
EarlyTracks	X	X	X		X					X		X	X	X		X		
ExB Group	X	X	X										X	X			X	
BECREATIVE	X	X	X		X		X			X								
PLsNER			X		X	X												X
mXS	X	X	X				X											
MoSTNER	X	X	X							X		X	X				X	
UKP	X	X			X								X					X

Table 5: Features used by systems. G = gazetteers, POS = part of speech, tok = token, NE-n = NE n-gram, cap = capitalization, lem = lemma, 1st = first word in span, last = last word in span, tok-n = token n-gram, #span = number of tokens in span, POS-n = POS n-gram, char = character-level, including affixes, n-grams, decompounding, WS = word shape, KW=keywords, SeC = semantic class, SiC = similarity clusters, WE = word embeddings

only show the best run submitted for each system, since our analysis has found that the within-system variance across runs is quite small compared to the between-system variance. The table is sorted according to F_1 measure.

It is clearly visible that the systems fall into three tiers: one top tier (ExB, UKP) with F-Scores between 75 and 77; a middle tier (PLsNER, MoSTNER, Earlystracks, DRIM) with F-Scores between 69 and 72; and a third tier with lower F-Scores.

The overall winner is the ExB system. Its victory is mostly due to its excellent recall of almost 4 points higher than that of the next-best system, while its precision is close to, albeit above, the median. Overall, all systems have a considerably higher precision that recall. We interpret this as an indication of the important role of successful *generalization* from the training data to novel, potentially different test data. The systems that were most successful in this generalization were the overall most successful systems in the shared task. Conversely, the system with the highest precision, mXS, does not fare well overall precisely due to its comparatively low recall.

Impact of Methods. Following up on the analysis from Section 4.1, we observe that purely rule-based systems and systems relying heavily on gazetteer queries could not reach competitive performance. In line with general trends in the field, it seems to be beneficial to rather plug in rules, lists and language-specific extractors as features in a machine learning framework than using them verbatim. As for machine learning methods, simple classification approaches that do not exploit information about interdependencies among datapoints are substantially outperformed by CRFs and Neural Networks. See (Hänig et al., 2014) for a direct comparison between ME and CRF using the same features.

System	Precision	Recall	F_1
ExB	78.07	**74.75**	**76.38**
UKP	79.54	71.10	75.09
MoSTNER	79.20	65.31	71.59
Earlytracks	79.92	64.65	71.48
PLsNER	76.76	66.16	71.06
DRIM	76.71	63.25	69.33
mXS	**80.62**	50.89	62.39
Nessy	63.57	54.65	58.78
NERU	62.57	48.35	54.55
HATNER	65.62	43.21	52.11
BECREATIVE	40.14	34.71	37.23
Median	76.71	63.25	69.33

Table 6: Precision, Recall, and F_1 for Metric 1 on the test set (official ranking)

Impact of features. Building on the results of Section 4.2, we observe that the three best systems have a comparatively small overlap in features: their intersection contains gazetteer-based, POS-level and character-level features. While gazetteers and parts of speech are used by nearly all the participating systems, the character-level features warrant further exploration. The best system, ExB, used several character query-based features in order to find sequences that are characteristic for NE classes, e.g. *-stadt*, *-hausen* or *-ingen*, which are typical endings for German cities. The

System	Precision	Recall	F_1
ExB	78.85	**75.50**	**77.14**
UKP	80.41	71.88	75.91
PLsNER	78.09	67.31	72.30
MoSTNER	79.94	65.92	72.26
Earlytracks	80.55	65.16	72.04
DRIM	77.53	63.92	70.07
mXS	**81.21**	51.26	62.85
Nessy	64.34	55.31	59.48
NERU	63.61	49.16	55.46
HATNER	66.19	43.58	52.56
BECREATIVE	40.78	35.26	37.82

Table 7: Precision, Recall, and F_1 for Metric 2 (subtypes *base*, *deriv* and *part* collapsed

MoSTNER system used Morphisto (Schmid et al., 2004; Zielinski and Simon, 2008) in order to divide tokens into morphological units at character level, which also may have categorized NE specific affixes. These morphological features can be understood as contributing to the generalization aspect outlined above.

The same is true for the use of *semantic* generalization features, which also can be found in different realizations in each of the three best system. Each used at least one high-level semantic feature, such as *Similarity Clusters* or *Word Embeddings*, that were rarely used by other systems. These features are computed in an unsupervised fashion on large corpora and alleviate sparsity by informing the system about words not found in the training set via their similarity to known words – be it as clusters of the vocabulary (MoSTNER, ExB) or vector representations (UKP, PLsNER). The use of simple semantic generalization to improve recall for NER was demonstrated in previous work (Biemann et al., 2007; Finkel and Manning, 2009; Faruqui and Padó, 2010).

5.2 Analysis by "loose metric" (M2)

Table 7 shows the evaluation results for the Metric 2 which does not distinguish between label subtypes.

Our main observation regarding Metric 2 is that the results are very similar to Metric 1. The three tiers can be identified exactly as for Metric 1, and the ordering in Tiers 1 and 3 is in fact identical. The only reordering takes place in Tier 2, where the differences among systems are so small (<.5% F_1) that this is not surprising. In absolute terms, systems typically do between .5% and 1% F-Score better on M2 than on M1, an improvement equally spread between higher precision and recall scores. Our conclusion is that the subtypes do not constitute a major challenge in the data.

Given that the M2 (four-class) results are most comparable to previous work on four-class NER, it is interesting to note that the best results of this challenge are quite close to the best reported results on the other prominent German dataset, the CoNLL 2003 newswire dataset. It is a question of further work to what extent this is a glass ceiling effect connected to, e.g., annotation reliability.

5.3 Per-Level Analysis (M3)

Finally, Table 8 shows the results according to Metric 3, that is, separately for inner and outer level NEs.

Across all systems, we see a noticeably worse performance on second-level NEs: the best F_1 on first-level NEs is 79, the best one on second-level NEs is 49. The more general observation is that first- and second-level NEs behave substantially differently. On first-level NEs, precision and recall are fairly balanced for most systems, with a somewhat higher precision. This is reflected in the maximum values reached: 82 points precision and 77 points recall, respectively. On second-level NEs, precision tends to be much higher than recall for many systems, often twice as high or even more. The maximum values obtained are 70 points precision and 41 points recall.

Another interesting finding is that the overall best system, ExB, is the best system for first-level NEs by a margin of over 2% F_1 (79% vs. 77%). In contrast, it is merely the median system on second-level NEs (43%) and performs more than five points F_1 below the best system, UKP (49%). Among all systems, UKP performs most consistently across first- and second-level NEs, obtaining second place on both levels. On the second level, is closely pursued by the Earlytracks system which shows a very high precision on second-level NEs (70%) but is hampered by a low recall (37%), resulting on an overall F-Score of 48%.

It is an open question for future analysis to what extent the large differences between first-

	First-level NEs			Second-level NEs		
System	**Precision**	**Recall**	**F_1**	**Precision**	**Recall**	**F_1**
ExB	80.67	**77.55**	**79.08**	45.20	41.17	43.09
UKP	79.90	74.13	76.91	58.74	**41.75**	**48.81**
MoSTNER	79.71	67.74	73.24	69.14	36.12	47.45
Earlytracks	80.44	66.98	73.10	**70.00**	36.70	48.15
PLsNER	77.93	68.52	72.92	57.86	37.86	45.77
DRIM	77.27	65.93	71.15	64.78	31.07	41.99
mXS	**81.90**	53.63	64.81	51.67	18.06	26.76
Nessy	64.83	56.93	60.62	42.86	27.38	33.41
NERU	63.67	51.33	56.84	33.85	12.62	18.39
HATNER	72.88	44.14	54.98	24.81	32.04	27.97
BECREATIVE	40.14	37.60	38.83	0	0	0

Table 8: Precision, Recall and F_1 for Metric 3, computed separately for first-level NEs and second-level NEs. Systems ranked according to F_1 on first-level NEs.

and second-level NEs reflect actual differences in difficulty (i.e., embedded NEs are more difficult to capture) and to what extent they are simply a result of the substantially smaller number of training examples (compare Table 1).

5.4 Per-NE Type Analysis

Finally, Table 9 shows the F_1 scores of the three best systems on the four NE classes from the data. All systems show the same patterns: best performance on PERson, followed by LOCation, ORGanization and finally on OTHer. The differences between PERson and LOCation are nonexistant to small (2%) while they perform substantially worse on ORG and again substantially worse on OTH. Again, it is interesting to compare the two top systems, ExB and UKP: UKP does slightly better on PER and LOC, the two most frequent classes (cf. Table 1), while ExB excels significantly for the two minority classes ORG and OTH. This complementary behavior indicates that there is a potential for ensemble learning using these systems.

In this comparison of NE types, the same question arises as for the comparison of levels: to what extent are the results a simple function of training set sizes? It is definitely striking that the ranking of the NEs types in terms of performance corresponds exactly to the ranking in terms of training data (cf. Table 1). At the same time, there is also reason to believe that the NE categories ORGanization and, in particular, OTH, are much less internally coherent than PER and LOC and therefore more difficult to model.

	ExB	UKP	MoSTNER
PER	84.05	85.48	82.54
LOC	84.05	84.62	80.47
ORG	76.29	69.60	62.24
OTH	59.46	49.81	48.38

Table 9: Peformance by NE type for top systems (F1 according to M1, outer chunks)

5.5 Comparing systems

An open question at this point is to what extent the submitted systems are complementary: do they make largely identical predictions or not? Given that the methods that the systems use are quite diverse, a large number of identical predictions could indicate problems with the dataset. Conversely, highly complementary output presents an opportunity for ensemble and other system combination methods. Historically, the best CoNLL 2003 system was also an ensemble (Florian et al., 2003).

We first computed the overlap between the predictions of each pair of systems at the word level, i.e., for what portion of words the two systems predicted the same label. We excluded words where both systems predicted O. Only the overall best run of each system was considered. We included the gold standard as a pseudo system (GOLD).

The results are shown in Table 10. The overlap

	UKP	Nessy	BECREATIVE	GOLD	NERU	ExB	DRIM	mXS	MoSTNER	PLsNER	Earlytracks	HATNER
UKP	—	0.447	0.317	0.594	0.406	0.561	0.542	0.448	0.578	0.613	0.568	0.389
Nessy	0.447	—	0.316	0.419	0.406	0.457	0.503	0.441	0.465	0.466	0.487	0.446
BECREATIVE	0.317	0.316	—	0.292	0.286	0.316	0.333	0.312	0.343	0.344	0.343	0.299
GOLD	0.594	0.419	0.292	—	0.392	0.614	0.525	0.418	0.556	0.558	0.553	0.361
NERU	0.406	0.406	0.286	0.392	—	0.431	0.442	0.426	0.432	0.443	0.442	0.448
ExB	0.561	0.457	0.316	0.614	0.431	—	0.550	0.460	0.578	0.572	0.576	0.406
DRIM	0.542	0.503	0.333	0.525	0.442	0.550	—	0.506	0.574	0.572	0.605	0.481
mXS	0.448	0.441	0.312	0.418	0.426	0.460	0.506	—	0.491	0.499	0.503	0.486
MoSTNER	0.578	0.465	0.343	0.556	0.432	0.578	0.574	0.491	—	0.610	0.619	0.437
PLsNER	0.613	0.466	0.344	0.558	0.443	0.572	0.572	0.499	0.610	—	0.595	0.453
Earlytracks	0.568	0.487	0.343	0.553	0.442	0.576	0.605	0.503	0.619	0.595	—	0.447
HATNER	0.389	0.446	0.299	0.361	0.448	0.406	0.481	0.486	0.437	0.453	0.447	—

Table 10: Pairwise word-level overlap of system predictions

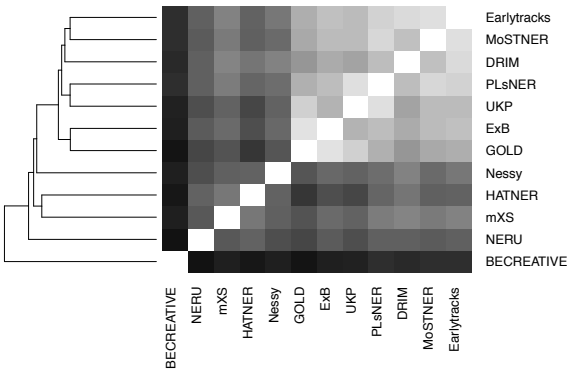


Figure 1: Heat map for pairwise system overlap

is relatively low: only a handful of comparisons yield an overlap of more than 0.5. We visualize the system comparisons as a heatmap in Figure 1. We see that BECREATIVE is very dissimilar to all other systems (it did not make any predictions for second-level NEs), while Earlytracks and MoSTNER have a comparatively high overall similarity to other systems (i.e., they produce a kind of "consensus" annotation). These two systems have also been clustered together, which may be related to the fact that they both use CRFs as their learning framework. Similarly, PLsNER and UKP, which are both based on neural networks, are also grouped together. The overall best system, ExB, has been grouped together with the gold standard.

Overall, these results look promising regarding future work on system combination. Without running a full-fledged analysis, we gauged the concrete potential by performing two simple analyses. The first one follows up on the per-level results from M3 (cf. Table 8), where we found that ExB and UKP show the best results for the first and the second level, respectively. Simply combining the ExB first level with the UKP second level yields a new best system with F_1=77.03 (M1), a further improvement of ΔF=.65 over ExB's previous result (cf. Table 6. The improvement notably is gained in precision (79.40 compared to 78.07) while recall stays about constant (74.79 compared to 74.75).

Finally, we computed an upper bound for the recall of an ensemble of the current systems. We performed this analysis because the fact almost all systems have a lower recall than precision (the best system has a recall of almost 75%, but the median is just at 63%) could be interpreted as an indicator that the corpus annotation is inconsistent or extremely difficult to recover automatically. However, when computing how many NE chunks in the gold standard are found by any of the systems, we determined that an oracle with access to all systems can cover 89.5% of the NE chunks. We take this result as an indication that there are no serious problems with the corpus, and that innovative strategies can hope to substantially improve over the current recall level.

6 Concluding remarks

In this paper, we have described the GermEval 2014 Named Entity Recognition shared task which extends the setup of traditional NER with morphologically motivated subtypes and embedded NEs.

The 11 submissions we received span a wide range of learning frameworks and types of features. The top systems appear to combine expressive machine learning techniques appropriate for the task (sequence classification and neural networks) with features that support intelligent generalization, notably encoding semantic knowledge.

The systems already achieve reasonable predictions on the dataset, in particular for precision-focused scenarios (median precision 76.7%, me-

dian recall 63.25%). At the same time, overlap in predictions between systems is surprisingly small, and system or feature combination may be able to further improve on the current results.

References

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. NoSta-D Named Entity Annotation for German: Guidelines and Dataset. In *Proceedings of LREC*, pages 2524–2531, Reykjavik, Iceland.

Chris Biemann, Claudio Giuliano, and Alfio Gliozzo. 2007. Unsupervised part of speech tagging supporting supervised methods. In *Proceedings of RANLP-07*, Borovets, Bulgaria.

Yulia Bobkova, Andreas Scholz, Tetiana Teplynska, and Desislava Zhekova. 2014. HATNER: Nested Named Entitiy Recognition for German. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Roman Capsamun, Daria Palchik, Iryna Gontar, Marina Sedinkina, and Desislava Zhekova. 2014. DRIM: Named Entity Recognition for German using Support Vector Machines. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Fabian Dreer, Eduard Saller, Patrick Elsässer, Ulrike Handelshauser, and Desislava Zhekova. 2014. BECREATIVE: Annotation of German Named Entities. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Manaal Faruqui and Sebastian Padó. 2010. Training and evaluating a German named entity recognizer with semantic generalization. In *Proceedings of KONVENS*, pages 129–133, Saarbrücken, Germany.

Jenny Rose Finkel and Christopher D Manning. 2009. Joint parsing and named entity recognition. In *Proceedings of HLT-NAACL 2009*, pages 326–334, Boulder, CO, USA.

Radu Florian, Abe Ittycheriah, Hongyan Jing, and Tong Zhang. 2003. Named entity recognition through classifier combination. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL*, pages 168–171. Edmonton, Canada.

Christian Hänig, Stefan Bordag, and Stefan Thomas. 2014. Modular Classifier Ensemble Architecture for Named Entity Recognition on Low Resource Systems. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Martin Hermann, Michael Hochleitner, Sarah Kellner, Simon Preissner, and Desislava Zhekova. 2014. Nessy: A Hybrid Approach to Named Entity Recognition for German. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Daniel Jurafsky and James H. Martin. 2000. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, Upper Saddle River, NJ, USA.

Jinseok Nam. 2014. Semi-Supervised Neural Networks for Nested Named Entity Recognition. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Nils Reimers, Judith Eckle-Kohler, Carsten Schnober, and Iryna Gurevych. 2014. GermEval-2014: Nested Named Entity Recognition with Neural Networks. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Helmut Schmid, Arne Fitschen, and Ulrich Heid. 2004. SMOR: A German Computational Morphology Covering Derivation, Composition and Inflection. In *Proceedings of LREC*, pages 1263–1266, Lisbon, Portugal.

Peter Schüller. 2014. MoSTNER: Morphology-aware split-tag German NER with Factorie. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Erik F Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-independent Named Entity Recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147, Sapporo, Japan.

Patrick Watrin, Louis de Viron, Denis Lebailly, Matthieu Constant, and Stéphanie Weiser. 2014. Named Entity Recognition for German Using Conditional Random Fields and Linguistic Resources. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Daniel Weber and Josef Pötzl. 2014. NERU: Named entity recognition for German. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Andrea Zielinski and Christian Simon. 2008. Morphisto – An Open Source Morphological Analyzer for German. In *Proceedings of the 7th International Workshop FSMNLP*, pages 224–231.

Modular Classifier Ensemble Architecture for Named Entity Recognition on Low Resource Systems

Christian Hänig **Stefan Bordag** **Stefan Thomas**
ExB Research & Development GmbH
Seeburgstr. 100
04103 Leipzig, Germany
[haenig|bordag|thomas]@exb.de

Abstract

This paper presents the best performing Named Entity Recognition system in the GermEval 2014 Shared Task. Our approach combines semi-automatically created lexical resources with an ensemble of binary classifiers which extract the most likely tag sequence. Out-of-vocabulary words are tackled with semantic generalization extracted from a large corpus and an ensemble of part-of-speech taggers, one of which is unsupervised. Unknown candidate sequences are resolved using a look-up with the Wikipedia API.

1 Introduction

Recognizing named entities in unstructured text in multiple languages and across different domains remains a challenging task. This can be gauged by the fact that for German the best Named Entity Recognition (NER) systems only achieve around 80% F1 (Faruqui and Padó, 2010). NER is even more difficult when resource limitations such as RAM usage or CPU time need to be taken into account, because then popular strategies such as simply using all possible character n-grams as features become infeasible. This is of particular importance when developing linguistic solutions for mobile platforms.

The relevant topics to cover when designing a NER system are which training data to use, which classifier to use and which features the classifier should be based on.

We present a NER system designed to minimize the impact of limited computational resources on the quality of the results and to maximize the cross-linguistic and cross-domain performance. This is implemented through a modular approach with complementary supervised components and unsupervised fall-back equivalents, ensuring adequate results even without part-of-speech (POS) annotated data.

2 Architecture of our solution

Our system consists of an ensemble of classifiers (see Section 2.1), list- (see Section 2.2) and pattern-based (see Section 2.3) annotators, and modules for the special treatment of out-of-vocabulary (OOV) words (see Section 2.4). Each module provides confidence scores for all annotations, which enables the ensemble to combine all candidate annotations to produce the most likely tag sequence (see Section 3).

2.1 Classifier-based annotation

Features typically encode aspects of either the target word or the surrounding words such as capitalization, part-of-speech or semantic information. In some languages, such as English, there are features which strongly indicate that the target word is a name, such as capitalization. Therefore NER systems for English typically achieve very good F1 scores of around 90% (e.g. 88.76% as reported by Sang and Meulder (2003)). In German, capitalization is used for all nouns and there are no such obvious features as strongly in-

dicative as English capitalization (Tkachenko and Simanovsky, 2012).

2.1.1 Features

We extract the following features for each of the tokens, usually in a 5-word-window around the target token:

Words Plain token strings

POS tags Tags obtained by a supervised tagger (Stanford Tagger as described by Toutanova et al. (2003)) and tags obtained by an unsupervised tagger based on *SVD2* as described by Lamar et al. (2010)

Word Shape Shape features based on Bikel et al. (1999) and shape features that are used by the Stanford NER (Finkel et al., 2005)

Semantic Classes We compute semantically similar words and cluster them as described by Gamallo and Bordag (2011), and use the resulting classes as features.

Additionally, we extract all n-grams of the target word (Finkel et al., 2005) and for compound words, use their components (e.g. *Berlin/Deutschland* leads to two additional word features: *Berlin* and *Deutschland*).

2.1.2 Classifier selection

Typically, classifier NER systems use either Conditional Random Fields (CRF) (Finkel et al., 2005), Maximum Entropy classifiers (ME) (Borthwick, 1999) or other machine learning methods. Apart from differing slightly in their generalization power, the classifiers also differ in training time, classification time and RAM usage. One interesting question is, how a probably slightly better classification method such as CRF compares to MaxEnt regarding runtime and memory consumption. One of the relevant differences is that ME classifiers tag each token individually, CRFs (and other sequence models like HMMs (Leek, 1997) and CMMs (Borthwick, 1999)) use adjacent words as well (Lafferty et al., 2001).

We experimented with three different classifiers: A collection of binary CRFs, a collection of binary MEs and a collection of improved binary MEs with an additional name boundary classification method. We trained them on the training data of GermEval 2014 (Benikova et al., 2014) with the features described in Section 2.1.1 and evaluated against the GermEval 2014 development data. Each of the classifiers was trained for each of the three NER categories *LOC*, *ORG* and *PER*. We additionally extended the ME classifier with a Boundary Detection algorithm (ME-BD) to overcome its weaknesses in sequence tagging. Therefore, we trained two ME classifiers: one for the left boundary and one for the right boundary, respectively. Each extracted entity is then extended employing both boundary classifiers until the most likely boundary has been detected.

Table 1 summarizes our results:

Classifier	Class	P	R	F
ME	LOC	0.854	0.569	0.683
ME	ORG	0.559	0.438	0.491
ME	PER	0.701	0.488	0.576
ME-BD	LOC	0.867	0.581	0.695
ME-BD	ORG	0.696	0.516	0.593
ME-BD	PER	0.893	0.609	0.724
CRF	LOC	0.856	0.632	0.727
CRF	ORG	0.793	0.502	0.615
CRF	PER	0.849	0.743	0.792

Table 1: M1 Scores for different classifiers / categories

Boundary detection significantly improves the performance of ME classifiers, especially for categories whose entities often consist of multiple tokens (e.g. *ORG* and *PER*). It took 8 hours to train the CRFs compared to 1 hour for the ME classifiers. Although CRFs provide clearly superior results in this experiment, it is obviously not feasible to train CRF models on mobile devices.

2.2 List-based annotation

We created entity lists for three NER categories and a catch-all *OTH* for unclassified NEs, as well as a number of subcategories for each (see Table 2 for a selection of these categories).

After crawling multiple freely available sources (e.g. OpenStreetMap[1] and Wikipedia[2]), we manually revised all extracted items. The main objective of this step is to reduce ambiguity to retain only high confidence items.

[1] http://www.openstreetmap.org/

[2] http://www.wikipedia.org/

The resulting lists are augmented with inflections, synonyms and abbreviations. We extracted all candidate items from a large word list computed for a crawled web corpus that are semantically or orthographically similar to the seed item. Finally, the suggested candidate items were manually revised and added to the entity lists.

NER category	subcategories
LOC	astronomical locations, castles, cities, continents, countries, highways, islands, lakes, mountains, (historical) regions, rivers, schools, seas, states, streets
PER	artists, historical persons, politicians, scientists, sportspersons, VIPs
ORG	aircraft / automobile / phone manufacturers, sports associations, cellphone providers, companies, financial institutions, musical bands, newspapers, organizations / associations, parties, politically motivated groups, radio channels, sports teams, television channels, universities / research institutes
OTH	airplane / automobile / cellphone models, currencies, historical events, products

Table 2: NER categories and selected subcategories

The list-based matching process shows a preference for longer matches over short ones (e.g. *FC Bayern München* supersedes *Bayern München*) and assigns a confidence score to each annotation. Confidence scores are estimated for each category separately based on an evaluation against our internal data sets.

2.3 Pattern-based annotation

Our pattern framework allows creation of almost arbitrary patterns, for example:

Suffix patterns If a word is uppercase and ends with *stadt* or *hausen* or *ingen* then annotate it as *LOC*.

Complex patterns If a word contains a dot followed by a top level domain and ends after the domain or is followed by a punctuation character then annotate it as *URL*[3] .

Sequence patterns If an uppercase word is followed by *AG* or *GmbH* or *Inc.* then annotate both words as *ORG*.

All patterns may be combined with specific exclusions to prevent incorrect high frequency words from being annotated (e.g. *Hauptstadt*[4]). Another heuristic that is used for lexicon matching also holds for pattern matching: long sequence matches supersede short matches.

2.4 Classification of Out-Of-Vocabulary words

We employ several strategies to cope with out-of-vocabulary words.

This includes the computation of both semantic generalizations (Faruqui and Padó, 2010) and syntactic generalizations of the words in the target data set (see Section 2.1.1) based on a large German web corpus (produced by our web crawler, consists of about 50M sentences).

We also compute a list of valid string transformations between categories. For each pair of words, a string transformation is computed (e.g. *Italien* to *italienische* is *lower-case(0) + -ische*). All obtained transformations are ranked according to their frequencies, pruned and manually revised. During classification these rules are applied to unknown words to transform them into possibly known words. This was applied on the source category *LOC* and the target categories *LOC*, *LOCderiv* and *LOCpart*.

Finally, we extract sequences of entity candidates (e.g. out-of-vocabulary uppercase words) and use the Wikipedia API to get more information about the candidates if category information is available in Wikipedia.

3 Classifier ensemble

The annotators finally vote on the joint output of the ensemble by sorting all the annotations of a sentence in descending order according to their

[3]URLs are mapped to OTH for this task.

[4]Means *capital city* and is a common noun.

confidence scores. Shorter annotations are discarded in case of overlaps.

The combiner then iterates over the ranked annotations and adds the annotation with the highest score as outer entity to the final tag sequence. If it overlaps with a higher ranked annotation of another type then it is added as inner entity instead. Any other types of overlaps are discarded. These steps are repeated until each of the annotations either has been added to the final tag sequence or has been discarded by the combination method.

3.1 Evaluation results

We created three models: a CRF model with unlimited resources (CRF; model size: 271MB), a low-resource CRF model (mCRF; model size: 41MB without technical compression) and a ME-BD model (ME-BD; model size: 159MB). The feature space of the low-resource CRF model was pruned significantly by removing n-grams and Stanford POS tags completely. Furthermore, the tremendous amount of token features is reduced to the 10k most frequent German words.

We trained all three models on the joint set of training and development data. The official evaluation scores obtained by evaluation against the test set are provided in Table 3:

Model	Metric	P	R	F
CRF	M1	0.781	0.748	0.764
CRF	M2	0.789	0.755	0.771
CRF	M3 outer	0.807	0.776	0.791
CRF	M3 inner	0.452	0.412	0.431
mCRF	M1	0.765	0.731	0.748
ME-BD	M1	0.786	0.734	0.759

Table 3: Official GermEval 2014 evaluation scores [5]

4 Conclusions

In our experiments we could verify that indeed CRFs produce better results compared to an improved ME (see Table 1), but the margin can be minimized by additionally applying further annotators (see Table 3).

We could also verify that it is possible to prune the feature space and thus, reduce resource consumption of NER models significantly to sizes which enable the NER system to be employed directly on mobile devices. Furthermore, the gap to the unrestricted CRF model (1.6%) is relatively small considering the huge amount of saved memory.

[5] See (Benikova et al., 2014) for metric definitions.

References

Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Pado. 2014. GermEval 2014 Named Entity Recognition: Companion Paper. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

D. M. Bikel, R. Schwartz, and R. M. Weischedel. 1999. An Algorithm that Learns What's in a Name. *Machine Learning*, 34(1-3):211–231.

A. Borthwick. 1999. *A Maximum Entropy Approach to Named Entity Recognition*. Phd, New York University.

M. Faruqui and S. Padó. 2010. Training and evaluating a German named entity recognizer with semantic generalization. In *Proceedings of KONVENS 2010*, pages 129–133.

J. R. Finkel, T. Grenager, and C. D. Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of ACL 2005*, pages 363–370.

P. Gamallo and S. Bordag. 2011. Is singular value decomposition useful for word similarity extraction? *Language Resources and Evaluation*, 45(2):95–119.

J. D. Lafferty, A. McCallum, and F. C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of ICML 2001*, pages 282–289. Morgan Kaufmann Publishers Inc.

M. Lamar, Y. Maron, M. Johnson, and E. Bienenstock. 2010. SVD and Clustering for Unsupervised POS Tagging. In *Proceedings of ACL 2010*, pages 215–219.

T. R. Leek. 1997. *Information Extraction Using Hidden Markov Models*. Master of science, University of California, San Diego.

E. F. T. K. Sang and F. De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of CoNLL 2003*, pages 142–147.

M. Tkachenko and A. Simanovsky. 2012. Named Entity Recognition : Exploring Features. In *Proceedings of KONVENS 2012*, pages 118–127.

K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of NAACL 2003*, pages 173–180.

GermEval-2014: Nested Named Entity Recognition with Neural Networks

Nils Reimers† **Judith Eckle-Kohler**†‡ **Carsten Schnober**†‡ **Jungi Kim**† **Iryna Gurevych**†‡
†Ubiquitous Knowledge Processing Lab (UKP-TUDA)
Department of Computer Science, Technische Universität Darmstadt
‡Ubiquitous Knowledge Processing Lab (UKP-DIPF)
German Institute for Educational Research
http://www.ukp.tu-darmstadt.de

Abstract

Collobert et al. (2011) showed that deep neural network architectures achieve state-of-the-art performance in many fundamental NLP tasks, including Named Entity Recognition (NER). However, results were only reported for English. This paper reports on experiments for German Named Entity Recognition, using the data from the GermEval 2014 shared task on NER. Our system achieves an F_1-measure of 75.09% according to the official metric.

1 Introduction

Neural network architectures using low-dimensional vector representations of words (word embeddings) as the (almost) only features have been shown to achieve state-of-the-art performance in many fundamental NLP tasks, such as POS tagging, parsing and Named Entity Recognition (NER) (Collobert et al., 2011). Word embeddings are distributed word representations that are learned in an unsupervised fashion. A distinguishing feature of word embeddings is their ability to capture properties of words at various levels, in particular semantic and morphosyntactic regularities: words with similar embeddings are semantically (or morphosyntactically) similar, i.e. they are close to each other in the low-dimensional embedding space (Mikolov et al., 2013).

Most previous NER shared tasks annotated named entities flatly (e.g. CoNLL (Tjong Kim Sang and De Meulder, 2003)) and ignored entities that are nested within each other, e.g., the top-level named entity "*Real Madrid*", an organization containing the nested location "*Madrid*". In contrast, the GermEval 2014 NER dataset also contains annotations of nested named entities (Benikova et al., 2014b). Besides the four main classes PERson, LOCation, ORGanization and OTHer, it also introduces for each main class the subtypes *-deriv* for adjectives referring to named entities (e.g. *euklidisch - Euclidean*) and *-part* for words only partly containing names (e.g. *deutschlandweit - Germany-wide*). The dataset is divided into a training set consisting of 24,000 sentences, a development set of 2,200 sentences and a test set of 5,100 sentences.

2 Named Entity Recognition using Neural Networks

Collobert et al. (2011) propose a unified neural network architecture that can be applied to various natural language processing tasks. The presented deep neural network architecture uses only features based on minimal preprocessing: lowercased words, capitalization of the words, part-of-speech and a small gazetteer of known named entities. The input sentence is fed into the architecture and several layers of abstractions are learned.

The first layer is a *lookup operation* which maps each word and its associated features (POS etc.) to a d-dimensional vector. The second layer

makes the assumption that the named entity tag of a word can be predicted from its neighboring words. The vectors from the lookup operation for the target word and the neighboring words are concatenated and fed through an affine transformation followed by a non-linear activation function like the hyperbolic tangent function.

There are two different approaches for the last layer of the network, depending on whether the *isolated tag criterion* or the *sentence tag criterion* is used. For the *isolated tag criterion*, each word in the sentence is considered independently. The probabilities of the different tags for each word are computed by the softmax-function.

The *sentence tag criterion* optimizes the label sequence over the entire sentence. Tag probabilities from each window are concatenated and the dependencies between tags are factored into the model by learning initial probabilities and transition probabilities between tags. The Viterbi algorithm is used during inference. Collobert et al. (2011) use the more expressive IOBES-tagging scheme in their experiments. It uses an S-tag to mark single word named entities and B-, I- and E-tags to mark the first, the intermediate and the last word of a multi-word named entity.

We address nested named entities by training two independent neural networks. The first one detects top-level named entities and the second one detects nested named entities. The neural network for the nested named entities is trained only on top-level named entities that span over two or more words. At inference time, the top-level model is applied first, and its classification result is fed as an additional feature into the model for nested named entities.

3 Word-Embeddings

Word embeddings are a representation of words in a dense vector space (Bengio et al., 2003). They serve as the main feature for our models and can be learned from unannotated text data.

We used the following six corpora with a total of 116 million sentences to pre-train the word embeddings: German Wikipedia, the Leipzig Corpora Collection (Biemann et al., 2007), the SDeWac corpus (Faaß and Eckart, 2013), the print archive of *Spiegel*[1], the print archive of *ZEIT*[2], and the articles of *ZEIT Online*[3]. We used the Word2Vec tool presented by Mikolov et al. (2013) to compute the word embeddings from our training corpus.

Apart from tokenization, we performed the following pre-processing steps: Numbers are substituted by the special token `0`, diacritics are removed, except for German umlauts. All tokens are lowercased; the semantics of capitalization in the German orthography is captured by the capitalization feature (cf. section 4) instead.

Decompounding could significantly increase the performance of named entity recognition, especially for *-part* named entites. Our system uses only a naïve decompounding strategy for out-of-vocabulary words. In case a word cannot be found in the vocabulary, we split it along non-alphabetic characters (e.g. hyphens or slashes). We then replace the word by the first part which can be found in the vocabulary.

4 Additional Features

We designed several features which we assume to be helpful for the task of tagging named entities.

Capitalization: A feature to cover the information whether the word is all uppercase, the initial character is uppercase or if any succeeding character is uppercase.

Hyphen-Decompound: This feature splits words with a hyphen and adds the word embedding for the first part of the splitted word.

POS: The POS-tags as assigned by TreeTagger (Schmid, 1995).

Gazetteer: A feature to cover the information if the word appears in various gazetteers with named entities which can freely be found on the internet. Most notably the provided gazetteers from (Tjong Kim Sang and De Meulder, 2003) and a city and country list by GeoBytes[4]. Additionally, we compiled a gazetteer for person names and locations based on the corresponding Wikipedia categories. Our gazetteers contains around 311,000 person names, 90,000 locations,

[1] `http://www.spiegel.de/spiegel/print/`
[2] `http://www.zeit.de/2014/index`
[3] `http://www.zeit.de/index`
[4] `http://www.geobytes.com/freeservices.htm`

	Pr	Re	F_1
STC	78.5%	69.1%	73.5%
STC+Hyphen	79.8%	71.4%	75.4%
STC+POS	78.8%	71.2%	74.8%
STC+POS+Hyphen	80.1%	72.1%	75.9%
STC+Gazetteer	79.0%	71.2%	74.9%
STC+Wikipedia	78.8%	71.6%	75.0%
STC+All Features	80.4%	74.1%	77.1%

Table 1: Performance for the *sentence tag criterion* (STC) and different hand-crafted features. Scores are computed for the top-level named entities on the GermEval 2014 test set.

3,800 organizations and 3,600 other named entities.

Wikipedia-Definition: A feature that uses the German Wikipedia as an external knowledge base. In contrast to (Kazama and Torisawa, 2007), we used the Mate dependency parser[5] to process the first sentence and from all nouns that are positioned after the root verb, we selected the one with the shortest path to the root.

5 Evaluation

The GermEval 2014 shared task is evaluated using precision, recall and F_1-measure. We have a true positive if we have an exact match on the span and an exact match on the assigned label. The offical metric for the shared task (Benikova et al., 2014a) also takes the level for an assigned label into account. This leads to some counter-intuitive behavior. For example, for the nested named entity *[[Fraunhofer]$_{ORG}$ FIT]$_{ORG}$*, a model that does not return any named entity is scored better than a model that returns only the nested named entity *Fraunhofer*. The latter model would place the tag for *Fraunhofer* on the first level and thus it would be considered a misclassification, resulting in a lower precision for this model. We provide results for a level-independent evaluation in section 5.2.

5.1 Separate Evaluation of top- and nested-level

Optimizing globally the label sequence over the entire sentence for the top-level named entities has a major impact on the performance of our system. Using no other features than the word-embeddings and the capitalization of the word, our system achieves an F_1-measure of F_1=69.9% for the *isolated tag criterion* and F_1=73.5% for the *sentence tag criterion*. We experimented with the IOB2- as well as with the IOBES-tagging scheme, but the difference was below 0.1% in F_1-measure. The nested named entities were covered by training a second, independent neural network. Our networks use a window size of 5, a decreasing learning rate between 0.1 and 0.01 and 150 hidden units.

Top-Level NE				
	#	Pr	Re	F_1
PER	1639	89.0%	84.7%	86.8%
PERderiv	11	–	0%	0%
PERpart	44	35.3%	13.6%	19.7%
LOC	1706	84.8%	83.8%	84.3%
LOCderiv	561	81.1%	88.8%	84.8%
LOCpart	109	77.8%	38.5%	51.5%
ORG	1150	71.8%	68.8%	70.3%
ORGderiv	8	–	0%	0%
ORGpart	172	70.6%	55.8%	62.3%
OTH	697	61.6%	43.3%	50.8%
OTHderiv	39	82.6%	48.7%	61.3%
OTHpart	42	63.6%	16.7%	26.4%
Nested NE				
PER	82	44.8%	31.7%	37.1%
LOC	210	58.0%	51.9%	54.8%
LOCderiv	159	68.1%	48.4%	56.6%
ORG	41	42.9%	7.3%	12.5%

Table 2: Number of named entities (#), Recall (Re), Precision (Pr) and F_1-measure for the differend named entity classes. Scores are for the test dataset using all features. Our model found none of the nested named entities with the classes PERderiv (#4), PERpart (#4), LOCpart (#5), ORGderiv (#1), ORGpart (#1), OTH (#7) or OTHpart (#1).

Table 1 gives an overview of the impact of the different features. By using POS-tags and the Hyphen-feature, we can increase the F_1-measure for the top-level named entities by 2.4% to F_1=75.9%. Adding external knowledge resources increases the score further by 1.2% to F_1=77.1% for the top-level named entities.

We can observe a large difference in F_1-measure for the different named entity classes. While for PER, our model achieves an F_1-measure of around 87%, we only achieve an F_1-

[5] http://code.google.com/p/mate-tools/

measure of 51% for OTH. Analyzing the data shows that OTH-named entities are often especially hard, for example titles of books or songs, and appear much less coherent than other classes.

5.2 Level-Independent Evaluation

Combining the scores for the top-level and the nested-level, our model achieves an F_1-measure of 75.1%. However, as noted above, the separate evaluation of top- and nested-level leads to some counter-intuitive behavior. When neglecting the level and only validating the span and the correct label, the F_1-measure for the same model is F_1=78.0%. This shows that in several cases our model finds only the nested named entity and not the corresponding top-level named entity.

Neglecting the level also allows to use an approach that learns the short named entities first, followed by the longer ones. With the proposed level-dependent evaluation, such an approach would be evaluated much worse because several named entities would probably be placed on the wrong level and would be considered as a misclassification. We therefore argue that future named entities evaluations should be level-independent.

6 Conclusion

We adapted the approach of Collobert et al. (2011) to German using the GermEval 2014 dataset. Without external resources, we achieve an F_1-measure of 75.9% on the test set for the top-level named entities. Adding gazetteers and knowledge extracted from the German Wikipedia increases the performance to 77.1% for the top-level named entities. Combined with the performance for the nested-level, we achieve an overall F_1-measure of 75.1% in the offical metric. When neglecting the two levels, and solely evaluating the correct span and the correct label, the performance of our model is 78.0%.

Acknowledgement

This work has been supported by the Volkswagen Foundation as part of the Lichtenberg-Professorship Program under grant No. I/82806, by the German Federal Ministry of Education and Research (BMBF) under the promotional reference 01UG1110D (DARIAH-DE), and by the Leibniz Association as part of the SAW project "Children and Their World".

References

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Pado. 2014a. GermEval 2014 Named Entity Recognition: Companion Paper. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014b. NoSta-D Named Entity Annotation for German: Guidelines and Dataset. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland.

Chris Biemann, Gerhard Heyer, Uwe Quasthoff, and Matthias Richter. 2007. The Leipzig corpora collection - monolingual corpora of standard size. In *Proceedings of Corpus Linguistics*, Birmingham, UK.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa. 2011. Natural Language Processing (almost) from Scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Gertrud Faaß and Kerstin Eckart. 2013. SdeWaC - A Corpus of Parsable Sentences from the Web. In *Language Processing and Knowledge in the Web*, pages 61–68. Springer.

Jun'ichi Kazama and Kentaro Torisawa. 2007. Exploiting Wikipedia as External Knowledge for Named Entity Recognition. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 698–707.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Helmut Schmid. 1995. Improvements In Part-of-Speech Tagging With an Application To German. In *Proceedings of the ACL SIGDAT-Workshop*, pages 47–50, Dublin, Ireland.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL-2003*, pages 142–147, Edmonton, Canada.

MoSTNER: Morphology-aware Split-Tag German NER with Factorie

Peter Schüller
Computer Engineering Department,
Faculty of Engineering, Marmara University, Istanbul, Turkey
peter.schuller@marmara.edu.tr

Abstract

MoSTNER is a German NER system based on machine learning with log-linear models and morphology-aware features. We use morphological analysis with Morphisto for generating features, moreover we use German Wikipedia as a gazetteer and perform punctuation-aware and morphology-aware page title matching. We use four types of factor graphs where NER labels are single variables or split into prefix (BILOU) and type (PER, LOC, etc.) variables. Our system supports nested NER (two levels), for training we use SampleRank, for prediction Iterated Conditional Modes, the implementation is based on Python and Factorie.

1 Introduction

Various Named Entity Recognition (NER) methods have been developed over time (Nadeau and Sekine, 2007) and currently many state-of-the-art systems rely on variations of Conditional Random Fields (CRF) (Sha and Pereira, 2003), with modifications that step away slightly from the Linear-Chain property, for example Skip-Chains (Sutton and McCallum, 2004), other non-local dependencies (Finkel et al., 2005), and Skip-Grams (Passos et al., 2014). Krishnan and Manning (2006) furthermore described an approach where two layers of CRFs are used to improve predictions of a single level of NER labels.

In the GermEval2014 competition for German nested NER several novel challenges needed to be addressed: German capitalization is not a useful feature as in English: adjectives and adverbs derived from names are not capitalized, while all nouns are capitalized; the rich morphology of German creates large noun compounds and makes Gazetteer usage challenging (these only contain the citation form); and nested NER is more challenging than single-level NER.

We next describe features used in the MoSTNER system, four variations of statistical models (some differ from linear-chain CRF quite much), learning and prediction methods, and performance on the GermEval2014 development set.

2 Features and Gazetteer Matching

We use most of the features that are well-known for English NER: token with simplified digits, POS-tag, shape, 4-letter token prefix and suffix, set of tokens in a left/right window of 1 to 4 tokens, POS-bigrams, and token/POS features shifted up to 2 tokens to left and right. POS-tagging was done with the Stanford tagger (Toutanova et al., 2003), moreover we use similarity-clustering using Clark's (2003) software with 400 clusters and 2 training iterations on 10M sentences (263M tokens) from the SdeWaC (Faaß and Eckart, 2013) corpus.

Novel features we added are the following:

- features based on morphological analysis with Morphisto (Schmid et al., 2004; Zielinski and Simon, 2008),
- German Wikipedia categories based on morphology-aware page title matching, and

- POS-bigram of the tag before and after the current token.

For each token, Morphisto generates a list of analyses that contains a sequence of token parts, analyzed as stems and morphological tags.

For example the token 'Presseberichten' ('news reports') obtains 15 analyses, one of them is 'Presse[NN]Bericht[+NN,Masc,Dat,Pl]'. We reduce these analyses by stripping off gender, case, and number morphological tags, and eliminating duplicates. For the above example this yields three analyses: 'Presse[NN]Bericht[+NN]', 'Presse[NN]be[Pref]richten[V,Suff,+NN]', and 'Presse[NN]berichten[V,Suff,+NN]'. From this reduced set of analyses and tags we create 10 distinct feature sets as follows:

- first/last/all stems of the token,
- all tags of the first/last/all token parts, and
- combinations of first/last stems in the left/right window of four neighbor tokens.

As Gazetteer we use German Wikipedia (dump from 20.3.2014) where we perform matching on page titles and redirection pages. Morphology-awareness is achieved by matching only a part of the input sequence (up to 3 characters from the end) in the Wikipedia database and then verifying all results against a regular expression built from the input that allows certain changes to the input sequence with the goal of transforming the input into its citation form: e.g., by stripping a final 's' we can transform genitive 'Maria-s' into nominative 'Maria', or by stripping final 'en' and allowing a Vovel to be added we can allow 'Kont-en' (accounts) and 'Vill-en' (villas) to match their citation forms 'Konto' and 'Villa', respectively.

From those Wikipedia page titles that match the training corpus, we select 1016 page categories (all that are found at least 10 times). If a Wikipedia page title matches a given sequence of tokens in the input, we generate features corresponding to each selected category as follows:

- each token obtains a feature containing the category;
- each token obtains a feature containing the category and a corresponding BILU tag, depending whether it is the first, interior, last, or unique token matching the page title.

This is also done for all subtokens of a token that can be split on a '-' symbol, e.g., 'EU-Minister'.

Stack CRF model		Single split-tag model	
Factors	# Weights	Factors	# Weights
		$bias^p_y$, $bias^p_z$	$5{+}5{=}10$
$bias$	$2{\cdot}49 = 98$	$bias^t_y$, $bias^t_z$	$13{+}13{=}26$
$-$	$-$	$stack^p$, $stack^t$	$5^2{+}13^2{=}194$
		$mark^p_y$, $mark^p_z$	$5^2{\cdot}2{=}50$
		$mark^t_y$, $mark^t_z$	$13^2{\cdot}2{=}338$
		$combo_y$	$5^2{\cdot}13^2{=}4225$
$mark$	$2{\cdot}49^2{=}4802$	$combo_z$	$5^2{\cdot}13^2{=}4225$
		$feat^p_y$, $feat^p_z$	$5{\cdot}2{\cdot}\lvert F\rvert$
$feat$	$2{\cdot}49{\cdot}\lvert F\rvert$	$feat^t_y$, $feat^t_z$	$13{\cdot}2{\cdot}\lvert F\rvert$
total	$4900{+}98{\cdot}\lvert F\rvert$	total	$9068{+}36{\cdot}\lvert F\rvert$

Table 1: Factors and number of weights in (i) a stack of 2 CRF models, and (ii) in a single model with split tags. Note that we use BILOU (5 possibilities) and GermEval uses 12 different NER types (PER, LOC, OTH, ORG, four derived, and four part subtypes).

Additionally we create the same features using a partial matching where any last three characters of the token sequence or the page title can be different. Partial matches are a separate feature set to allow the learning to assign different levels of confidence to partial and exact matches.

Moreover, if there is a pair of punctuation signs (e.g., between '"' and '"', between '(' and ')', and between '-' and '-') around 2 to 4 tokens, we copy all non-BILU Gazetteer features from first and last token to these tokens.

3 Factor Graph Layout(s)

We experimented with four statistical models. MoSTNER is implemented using Python (feature generation) and Factorie (training and prediction of statistical models). We train and predict using BILOU as suggested in (Ratinov, 2012). Figure 1 shows a Linear-Chain CRF for one level of NER labeling on the left side, and a model for labeling two levels of NER with split-tag variables on the right side. The most important characteristic of the split-tag model is that it splits each NER tag (e.g., 'B-POSderiv') into two variables: the BILOU prefix (e.g., 'B') and the NER type (e.g., 'POSderiv'). The idea is to connect the concerns of predicting BILOU with the concern of predicting NER types only where necessary.

Details of the model are as follows: prefixes

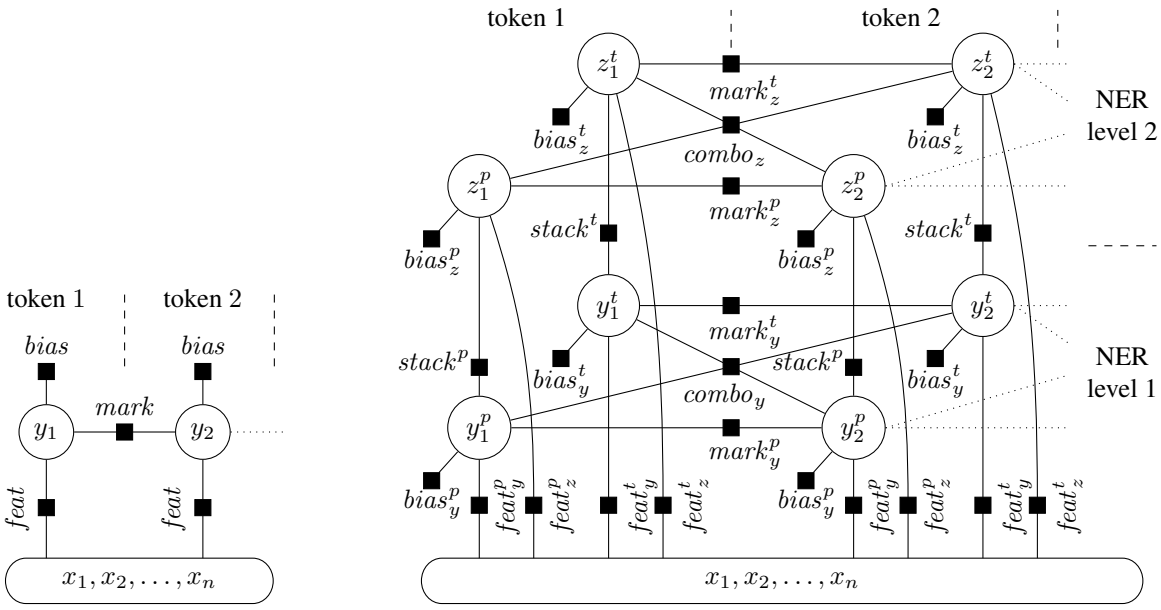


Figure 1: Linear-Chain CRF (left) and single-model split-tag factor graph (right).

and types obtain biases (factor $bias$) and they are connected via Markov-chains within their respective layers (factors $mark$), moreover the two NER levels are connected via factors $stack$ and each level has separate training weights (e.g., factors $feat_p^y$ vs $feat_p^z$ for prefix features). The only factor that relates prefix and type (for span consistency) is $combo$, and this factor does not connect levels.

As shown in Table 1, splitting the tags has the consequence that our single split-tag model obtains fewer weights to train compared with two stacked Linear-Chain CRFs that predict each level separately with one variable per tag. (This is due to the usually high number of features $|F|$.)

4 Experiments

We experimented with four types of models: stacking two Linear-Chain CRFs (Fig. 1 left), using a single split-tag model (Fig. 1 right), stacking two split-tag models (not depicted, imagine stacking two models containing only NER level 1 of Fig. 1 on the right) and using a single model that includes two CRFs stacked on top of each other (not depicted, imagine Fi.e 1 on the right without split tags). For stack models we first train a model to predict the first (outer) NER layer and then a model for the second layer that obtains the first level's predictions as additional features.

For the Linear Chain model we used Viterbi (exact inference) and update weights using the Adaptive Subgradient method by Duchi et al. (2010). The other three models contain cycles, hence exact training and inference methods are not available. We therefore train with SampleRank (Wick et al., 2009) using Gibbs Sampling and a temperature of 0.0001,[1] we update weights using MIRA (Crammer and Singer, 2003). For prediction we use Iterated Conditional Modes (2 iterations) (Besag, 1986). Other learning methods and parameters performed slightly worse.

Model	Notes	Level 1 P-R-F1(%)	Both Levels P-R-F1(%)
Stack-single	Fig. 1 left	76-71-**73.5**	75-69-**72.1**
Stack-split	not depicted	71-67-68.8	71-65-67.7
Single-single	not depicted	74-72-72.8	73-70-71.6
Single-split	Fig. 1 right	73-71-71.9	72-69-70.4

Table 2: GermEval2014 development set performance comparison (official, strict metric). Stack models consist of two separate models, one for each NER level, while single models predict both levels together.

5 Related Work and Conclusion

Faruqui and Padó (2010) described a German NER system with distributed similarity cluster-

[1] For more greedy training (thanks to Michael Wick).

ing and morphology-based features with a linear-chain CRF. MoSTNER additionally uses morphology for Gazetteer lookup and we experiment with more complex models. We did not consider parsing-based approaches as done by Finkel and Manning (2009) for English nested NER.

Performance of MoSTNER on the GermEval2014 (Benikova et al., 2014) development set is shown in Table 2: results indicate that the simplest solution (two Linear-Chain CRFs, one for each NER level) achieves the best prediction correctness. F1-scores on the test set of GermEval2014 are shown in the following table for all the metrics used in the competition.

Model	run	strict	loose	level 1	level 2
Stack-single	3	71.59	72.26	73.24	47.45
Single-split	2	69.18	70.17	70.59	43.80

Experiments with feature sets show that Morphisto features and partial Wikipedia matches decrease performance of the simple CRF, while they increase performance of other models. We plan to perform future work on these observations and publish the source code of MoSTNER.

Acknowledgments

We are grateful for help received from Factorie mailing list members. This work is supported by the Faculty of Engineering, Marmara University.

References

Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Pado. 2014. GermEval 2014 Named Entity Recognition: Companion paper. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Julian Besag. 1986. On the Statistical Analysis of Dirty Pictures. *Journal of the Royal Statistical Society Series B (Methodological)*, 48(3):259–302.

Alexander Clark. 2003. Combining Distributional and Morphological Information for Part of Speech Induction. In *Conference of the European Chapter of the Association for Computational Linguistics*, pages 59–66.

Koby Crammer and Yoram Singer. 2003. Ultraconservative Online Algorithms for Multiclass Problems. *Journal of Machine Learning Research*, 3:951–991.

John Duchi, Elad Hazan, and Yoram Singer. 2010. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. Technical report, University of California at Berkeley.

Gertrud Faaß and Kerstin Eckart. 2013. SdeWaC - A Corpus of Parsable Sentences from the Web. In *Language Processing and Knowledge in the Web*, pages 61–68.

Manaal Faruqui and Sebastian Padó. 2010. Training and Evaluating a German Named Entity Recognizer with Semantic Generalization. In *Verarbeitung natürlicher Sprache (KONVENS)*.

Jenny Rose Finkel and Christopher D Manning. 2009. Nested Named Entity Recognition. In *EMNLP*, pages 141–150.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *ACL*, pages 363–370.

Vijay Krishnan and Christopher D Manning. 2006. An Effective Two-Stage Model for Exploiting Non-Local Dependencies in Named Entity Recognition. In *ACL/COLING*, pages 1121–1128.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon Infused Phrase Embeddings for Named Entity Resolution. In *CoNLL*.

Lev Ratinov. 2012. *Exploiting Knowledge in NLP*. Phd thesis, University of Illinois at Urbana-Champaign.

Helmut Schmid, Arne Fitschen, and Ulrich Heid. 2004. SMOR: A German Computational Morphology Covering Derivation, Composition, and Inflection. In *LREC*, pages 1263–1266.

Fei Sha and Fernando Pereira. 2003. Shallow Parsing with Conditional Random Fields. In *NAACL-HLT*, pages 134–141.

Charles Sutton and Andrew McCallum. 2004. Collective Segmentation and Labeling of Distant Entities in Information Extraction. In *ICML Workshop on Statistical Relational Learning*.

Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *NAACL-HLT*, pages 252–259.

Michael Wick, Khashayar Rohanimanesh, Aron Culotta, and Andrew McCallum. 2009. SampleRank: Learning Preferences from Atomic Gradients. In *NIPS Workshop on Advances in Ranking*, pages 69–73.

Andrea Zielinski and Christian Simon. 2008. Morphisto - An Open Source Morphological Analyzer for German. In *Workshop on Finite-State Methods and Natural Language Processing*, pages 224–231. IOS Press.

HATNER: Nested Named Entitiy Recognition for German

Yulia Bobkova, Andreas Scholz, Tetiana Teplynska, Desislava Zhekova
CIS, Ludwig Maximilians University, Munich
{Yulia.Bobkova,Andreas.Scholz,Tetiana.Teplynska,D.Zhekova}
@campus.lmu.de

Abstract

This paper describes our classification and rule-based attempt at nested Named Entity Recognition for German. We explain how both approaches interact with each other and the resources we used to achieve our results. Finally, we evaluate the overall performance of our system which achieves an F-score of 52.65% on the development set and 52.11% on the final test set of the GermEval 2014 Shared Task.

1 Introduction

Named Entity Recognition (NER) is currently one of the most interesting and promising topics in NLP. It is commonly viewed as a subtask of information extraction (Nagy T. et al., 2011) and is a basis for many important applications, such as Coreference Resolution and Sentiment Analysis. NER by itself is no trivial task and NER for German is even more challenging, as the amount of available manually annotated data is limited. Additionally, capitalization is usually an important feature for detecting NEs. However, as nouns are generally capitalized in German, the usefulness of the capitalization feature is diminished. The quality of a NER system also strongly depends on its domain, as a system tailored to one specific domain generally performs worse on other domains (Poibeau and Kosseim, 2001). In this paper, we present a hybrid approach to NER in the implementation of HATNER.

Section 2 gives an overview of other approaches to NER. In section 3, we go into detail about the system requirements. In section 4, we give a short overview of HATNER and in Sections 4.1, 4.2 and 4.3 we go into more detail about the system. In section 5, we present our results and discuss them accordingly. Finally, in section 6 we conclude the our work.

2 Related Work

One of the earliest systems, which originally was intended for the English language only, is GATE (Cunningham et al., 2011). GATE itself is a conglomeration of different tools for NLP. One of these tools is ANNIE (a Nearly-New Information Extraction System) also described in (Cunningham et al., 2003). ANNIE uses finite-state algorithms and the JAPE language for regular expressions, as well as several gazetteers. During ongoing development support for more languages was added, amongst them German.

Another interesting approach and one of the best for English available today is the Stanford Named Entity Recognizer. It is based on a Conditional Random Field classifier and performs particularly well on the categories person, organization and location.

Lastly, specifically for German, there is one of the few freely available NER systems developed by Faruqui and Padó (2010). It is based on the previously mentioned Stanford NER and includes semantic generalization information from large untagged German corpora. It is one of the best NER systems for German available today.

Unfortunately, most state-of-the-art NER systems have not been developed with nested NEs in mind, which was newly initiated by the GermEval 2014 Named Entity Recognition Shared Task.

http://gate.ac.uk

http://gate.ac.uk/sale/tao/splitch6.html\#chap:annie

http://nlp.stanford.edu/software/CRF-NER.shtml

3 System Requirements

HATNER was specifically developed for the context of the GermEval 2014 Shared Task. The shared task specifies four main categories of entities to be recognized: person (PER), location (LOC), organization (ORG) and other (OTH), where OTH contains categories such as time, date, currency, religion and more. Each word or group of words in the data can qualify for any of these four categories, or none. For each of these four main categories, there also exists a part and derivative subcategory (labeled i.e. PERpart or PERderiv). Detailed information as to when a NE qualifies as part or derivative of a main category and the main categories themselves are specified by the NE annotation guidelines (Benikova et al., 2014). In short, one can define an entity as belonging to the part subcategory, if only a part of the NE belongs to a specific category, such as "Wembley-Tor", where Wembley is a LOC. The derivative category on the other hand mostly encompasses morphologically modified NEs, such as "Berliner" (as in: a citizen of Berlin, LOCderiv).
This results in a total of 12 possible categories for a NE. However, the aim of the shared task is not only to find NEs, but also to find NEs within said NEs. Hence, in a sentence like "Ich lese 'Das Tagebuch der Anne Frank'." there are two NEs: "Das Tagebuch der Anne Frank" (OTH), as well as "Anne Frank" (PER). Figure 1 shows an example of the annotation format as given in (Benikova et al., 2014). The second column depicts the word itself, followed by the NE tag for the first NE level and the NE tag for the nested NE level respectively. A tag starting with a B indicates the beginning of a NE. I indicates the inside of a NE and O the outside.

#	http://de.wikipedia.org/wiki/Manfred_Korfma		
1	Aufgrund	O	O
2	seiner	O	O
3	Initiative	O	O
4	fand	O	O
5	2001/2002	O	O
6	in	O	O
7	Stuttgart	B-LOC	O
8	,	O	O
9	Braunschweig	B-LOC	O
10	und	O	O
11	Bonn	B-LOC	O
12	eine	O	O
13	große	O	O
14	und	O	O
15	publizistisch	O	O
16	vielbeachtete	O	O
17	Troia-Ausstellung	B-LOCpart	O
18	statt	O	O
19	,	O	O
20	,,	O	O
21	Troia	B-OTH	B-LOC
22	-	I-OTH	O
23	Traum	I-OTH	O
24	und	I-OTH	O
25	Wirklichkeit	I-OTH	O
26	"	O	O
27	.	O	O

Figure 1: Example of a tagged sentence in the final output file. (Benikova et al., 2014)

4 System Overview

Classification systems are generally more robust to change than rule-based systems and perform fairly well with an adequate feature set. However, they heavily rely on a large and qualitatively annotated training set. On the other hand, rule-based systems are very susceptible to changes and very time consuming to establish, but can better be tailored to specific needs. For these reasons, we propose a classification approach as the core of our system, which we also combine with a set of handcrafted rules specifically targeting the distinct NE types.

4.1 Preprocessing and Postprocessing

In order to provide our classifier with as many useful features as possible, we preprocessed each sentence. This included noun phrase identification, lemmatization and part of speech (POS) tagging. For this, we used the Python programming language as well as the NLTK toolkit and the TreeTagger (Schmid, 1994; Schmid, 1999).
As for postprocessing, the most important task is to ensure a well formed output file. Other than rules, a classifier is not guaranteed to always start a recognized NE with a beginning tag, but could instead start with an inside tag. Our postprocessing ensured the correct opening of each NE. We tried several different approaches, such

https://www.python.org
http://www.nltk.org
http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger

as conservative processing (converting I to O), neutral processing (I to B), optimistic processing (tag the previous word as beginning of the same category) or intelligent processing (considering noun phrases and sentence structures when deciding how to proceed). For the final tagging process we used conservative postprocessing as it provided the best results. Another step of postprocessing that we do is eliminating inner tags found by the second classifier which are not inside of any outer tag.

4.2 Classification

For the classification task, we use a maximum entropy classifier which is trained on the manually pre-tagged training set provided by GermEval.
We train two classifiers: one for the first NE level and the second one for the nested, NE level. Inbetween the classifier runs we perform a postprocessing step to ensure a well formed file for the second run. In order to achieve the best results, we devised and tested different features. The features of our final system are displayed in table 1.

For the second classifier, we use a subset of these features together with a feature which indicates whether an outer NE exists for the current token. The second run is also much more delicate. While the classifier is in fact encouraged to only tag tokens which were previously tagged as belonging to an outer NE, there is no guarantee for that. As we mention before, we compensate this with another post-processing step which handles inner tags which do not belong to an outer tag.

4.3 Rules

In the second part of HATNER, we specifically target areas the classifier had difficulties with, such as part and derivative forms of categories. With rules focusing on precision rather than recall, we intend to affect the results of the classifier as least as possible, while at the same time having a high confidence at actually improving or correcting a tag once all conditions of a rule had been met.
To keep the rules as specific as possible, it was not enough to use morphological and syntactic features only. We therefore created gazetteers for each of the four main categories. We extracted information from the German Wikipedia and also used the gazetteers available in the GATE system. Here, once again, German being the object of our studies turned out to be an added difficulty. Lists for the English language can easily be found, already available lists for German are scarce and inconsistent at best, non-existent at worst. Additionally, we need to detect which tokens may be part of a NE, so we lowercased the entries in the gazetteers, what led to the loss of information.

Feature	1st Cl.	2nd Cl.
The token itself	yes	yes
The POS tag of the token	yes	yes
The POS tag of the previous token	yes	yes
The lemma of the token	yes	yes
Whether the token is within a NP	yes	no
The history of tags of the sentence	yes	yes
Outer NE tags assigned to this token	no	yes

Table 1: Feature sets of the first ($\mathbf{1}^{st}$ Cl.) and second ($\mathbf{2}^{st}$ Cl.) classifier.

As for the gazetteers, we aimed at matching maximum length spans. However, during development, lists with less, but more specialised information performed better than large general lists. For example, after stripping down the names list to just common German and English names, we received much better results than with names from all over the world, as many of those tended to correlate with common, non-name words, in German.

5 Results and Evaluation

Table 2 shows the general results of the HATNER system, whereas table 3 shows the results of the part and deriv subcategories for each of the four main categories. We report results on the development set.

Setup	Chunks	Prec.	Rec.	F1
Classifier	Outer	71.26	44.98	**55.15**
	Inner	26.94	37.74	**31.43**
	Combined	64.59	44.44	**52.65**
Classifier + Rules	Outer	60.57	46.14	52.38
	Inner	19.12	30.66	23.55
	Combined	54.61	45.00	49.34

Table 2: General results of the system.

As can be seen in table 2, the final score of the classifier and rules combination is actually

http://de.wikipedia.org

Category	Classifier only	Classifier & Rules
LOCderiv		
outer	**75.43**	68.42
inner	**55.45**	19.69
LOCpart		
outer	29.51	**36.00**
inner	0.0	0.0
ORGderiv		
outer	0.0	0.0
inner	0.0	0.0
ORGpart		
outer	19.80	**55.63**
inner	0.0	0.0
OTHderiv		
outer	**46.15**	42.86
inner	0.0	0.0
OTHpart		
outer	10.53	**18.18**
inner	0.0	0.0
PERderiv		
outer	0.0	0.0
inner	0.0	0.0
PERpart		
outer	**10.53**	6.45
inner	0.0	0.0

Table 3: Subcategory results of the system.

performing worse than the classifier on its own. Interestingly enough, the classifier also performs better on nested NEs than the combined system. On the other hand, rules do improve some of the subcategories we actually designed them to improve. Table 3 shows that, while the derivative category seems to pose the most difficulties for either system, rules were able to compensate some of the weaknesses of the classifier in most of the part categories.

HATNER achieved 52.11% on the final test set based on the combined evaluation setting from table 2 (being M1, the official metric used by the task).

6 Conclusion

The paper presented the participation of our system at the GermEval 2014 Named Entity Recognition Shared Task for German. The results HATNER achieved on the development set indicate two facts: First, the combination of the classifier and the rules is worse than the classifier by itself. Second, rules are able to improve certain areas if tailored specifically to these areas. This leads us to believe, that, while this implementation of a combined system might have failed, it generally is possible and desirable. In our eyes, the key to achieving a combined system which actually performs better is to specialise rules even more. This would decrease the negative effect on the work of the classifier, while increasing the positive effects on the areas they would be designed to improve.

References

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. NoSta-D Named Entity Annotation for German: Guidelines and Dataset. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Cristian Ursu, and Marin Dimitrov. 2003. Developing Language Processing Components with GATE (a User Guide). http://gate.ac.uk, February.

Hamish Cunningham, Diana Maynard, and Kalina Bontcheva. 2011. *Text Processing with GATE*. Gateway Press CA.

Manaal Faruqui and Sebastian Padó. 2010. Training and Evaluating a German Named Entity Recognizer with Semantic Generalization. In *Proceedings of KONVENS 2010*, page 129. Semantic Approaches in Natural Language Processing.

István Nagy T., Gábor Berend, and Veronika Vincze. 2011. Noun Compound and Named Entity Recognition and their Usability in Keyphrase Extraction. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 162–169, Hissar, Bulgaria, September. RANLP 2011 Organising Committee.

Thierry Poibeau and Leila Kosseim. 2001. Proper Name Extraction from Non-Journalistic Texts. In *In Computational Linguistics in the Netherlands*, pages 144–157.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

Helmut Schmid. 1999. Improvements in Part-of-Speech Tagging with an Application to German. In Susan Armstrong, Kenneth Church, Pierre Isabelle, Sandra Manzi, Evelyne Tzoukermann, and David Yarowsky, editors, *Natural Language Processing Using Very Large Corpora*, volume 11 of *Text, Speech and Language Processing*, pages 13–26. Kluwer Academic Publishers, Dordrecht.

DRIM: Named Entity Recognition for German using Support Vector Machines

Roman Capsamun, Daria Palchik, Iryna Gontar, Marina Sedinkina, Desislava Zhekova
CIS, Ludwig Maximilian University, Munich
{R.Capsamun,D.Brykova,Iryna.Gontar,Marina.Sedinkina,D.Zhekova}
@campus.lmu.de

Abstract

This paper[1] describes the DRIM Named Entity Recognizer (DRIM), developed for the GermEval 2014 Named Entity (NE) Recognition Shared Task.[2] The shared task did not pose any restrictions regarding the type of named entity recognition (NER) system submissions and usage of external data, which still resulted in a very challenging task. We employ Linear Support Vector Classification (Linear SVC) in the implementation of SckiKit,[3] with variety of features, gazetteers and further contextual information of the target words. As there is only one level of embedding in the dataset, two separate classifiers are trained for the outer and inner spans. The system was developed and tested on the dataset provided by the GermEval 2014 NER Shared Task. The overall strict (fine-grained) score is 70.94% on the development set, and 69.33% on the final test set which is quite promising for the German language.

1 Introduction

Named Entity Recognition aims to detect and classify nominal phrases into predefined categories such as organization, person, location and other. So far, mostly flat NEs were the target of identification (Benikova et al., 2014), which has been changed for GermEval 2014. This task is very important for many NLP challenges, such as information retrieval, speech processing, data mining, question answering, automatic summarization etc.

Most of the research in this field has been carried out for English with systems achieving considerably high levels of recall (97%) and precision (95%) (Mikheev et al., 1998; Stevenson and Gaizauskas, 2000). Though those results are substantial, the situation for other languages, especially for German, seems to be different.

Rules that are applied to English are not always useful for German. For example, in German not only NEs, but all the nouns are capitalized. In distinction to English, German adjectives such as "deutsch" are not to be capitalized. In comparison to English, German has higher morphological complexity, most productive type of which are compounds that are not found in a dictionary, for example, *AXA-Kunde, ADAC-Mitglied, Victoria-Agentur*. Except compounds, there are also derivations containing NEs, for instance, *die Deuscthen, die Bremer Staatsanwaltschaft*. The GermEval 2014 Shared Task sets as a goal the identification of both levels. A big obstacle is that existing training datasets for German are hindered by license problems. Also, there are not many open source NER taggers for German that perform at high levels of accuracy.

Because of these facts, proper identification and classification of NEs in German are very crucial and set a big challenge to the NLP research.

In Section 2, we describe related NER research.

[2]https://sites.google.com/site/germeval2014ner

[3]http://scikit-learn.org/stable

In Section 3, the data sets and the tagset provided by the GermEval 2014 NER Shared Task are presented, while in Section 4, we give an overview of Linear SVC. Following, we focus on the features that were used (see Section 5). Finally, we present our results on the development set provided by GermEval 2014 in Section 6, and in Section 7 we summarize our work and give suggestions for future directions.

2 Related Work

Since the Sixth Message Understanding Conference (MUC-6)[4], NER has become a well-established task of information extraction systems. MUC was initiated and financed by the Defense Advanced Research Projects Agency[5] to encourage the development of new and better methods of information extraction. Such competitions aimed at establishing frameworks for the proper and objective evaluation of various systems performing the same task (providing datasets and scoring possibilities).

For NER different approaches have been developed so far. There is a freely available Java implementation of a Named Entity Recognizer for English, namely Stanford NER.[6] As for the other languages, in particular German, one of the most significant works were presented by Faruqui and Padó (2010). Their German NER tagger has been trained on the CoNLL 2003 Shared Task[7] (Tjong Kim Sang and De Meulder, 2003) train set and uses semantic generalization information from two large German corpora, namely the HGC (Stuttgart University Newspaper Corpus) and deWac (the .de top-level domain "web as corpus"). Since 2010, this system is among the best NER systems for German with precision of 88.0% and recall of 72.9% (Faruqui and Padó, 2010).

There are also other machine learning systems for German NER. For example, Rössler (2004) similar to Faruqui and Padó (2010) uses resources with lexical knowledge from untagged corpora, reaching 78% recall and 71% precision (Rössler, 2004).

Rule-based approaches are also used for NER. The manually created rule-based system elaborates a set of patterns to accurately recognize and tag NEs (Volk and Clematide, 2001). They have reached 86%(recall) and 92%(precision). Another well-known rule-based system is Syntactic Constraint Parser (SynCoP), that is based on TAGH-morphology and gazetteers (Geyken and Schrader, 2006). Using the largest annotated corpus in the molecular biology domain, namely GENIA, the NER from Shen et al., (2003) trained a Hidden Markov model over the inner named entities, and then used a rule-based approach to identify the named entities containing the inner entities (Shen et al., 2003).

In our work, we implement a machine-learning approach with two separate linear SVM classifiers which are trained for the outer and nested spans of the NEs present in the GermEval 2014 dataset.

3 Named Entity Data and Tagset

The GermEval 2014 NER Shared Task provides a new dataset. This data was sampled from the German Wikipedia and News Corpora as a collection of citations. The dataset covers over 31,000 sentences corresponding to over 590,000 tokens. It is publicly available for download[8] under the permissive CC-BY license. The data has been annotated by two native speakers according to the semantic-based guidelines (Benikova et al., 2014). The entities from the dataset are to be classified in four main categories (*PER* – person; *ORG* – organization; *LOC* – location; *OTH* – other) with three subclasses (*main*, a NE comprises the full span; *part*, a NE takes only part of the span and *deriv*, the span is a derivation of a NE).

As for the format, each sentence is encoded as one token per line, with information provided in tab-separated (TSV) columns. The first column contains the token number within the sentence. The second column is the token itself. Name spans are encoded in the BIO-scheme (begin-

[4] http://cs.nyu.edu/cs/faculty/grishman/muc6.html

[5] http://www.darpa.mil

[6] http://nlp.stanford.edu/software/CRF-NER.shtml

[7] http://www.cnts.ua.ac.be/conll2003/ner/

[8] https://sites.google.com/site/germeval2014ner/data

inside-outside). An example of the data format used in this shared task can be seen in Table 1.

TokenId	Token	Outer	Inner
21	Troia	B-OTH	B-LOC
22	-	I-OTH	O
23	Traum	I-OTH	O
24	und	I-OTH	O
25	Wirklichkeit	I-OTH	O

Table 1: Example of the data format.

4 Linear Support Vector Machine

Support Vector Machines (SVMs) are set of supervised learning methods used for classification, regression and solving various pattern recognition problems. This state-of-the-art classification method was introduced in 1992 by Boser, Guyon and Vapnik (Boser et al., 1992). Even though it is a relatively new machine learning approach, SVMs are well known for their good generalization performance and efficiency in high dimensional spaces (Kudo and Matsumoto, 2001). In the field of NLP, SVMs are reported to have achieved high accuracy in text categorization without falling into over-fitting because of a large number of words taken as a feature (Kudo and Matsumoto, 2000). Linear SVC has also been used in DRIM. The model assumes that the data is linearly separable. Linear SVC implements "one-vs-the-rest" multi-class strategy, thus training class models.

5 Feature Description

The most significant role in Support Vector Machines (SVM) plays feature selection (Ekbal and Bandyopadhyay, 2008). As there is one level of embedded NEs, two different classifiers were trained for each layer of embedding (further called outer and inner span).

5.1 Outer Span

5.1.1 Morphological Features

This class of features includes the most informative characteristics such as the token itself, Part of Speech (POS) information, lemma, token suffix, prefix and root. Morphological features are very basic but at the same time significant features which we take as a baseline.

POS information and lemmas are obtained via the TreeTagger (Schmid, 1994; Schmid, 1995), developed by Helmut Schmid.[9] TreeTagger makes use of a decision tree to get more reliable estimates for contextual parameters. This method has resulted in a higher accuracy than a standard trigram tagger (Schmid, 1994).

Token suffix, prefix and root are also informative features for NER. Considering the variety of German morphological entities we use a fixed length (four characters) of token suffix or prefix in a respective suffix/prefix feature. This length is very useful in detecting German suffixes, like *-land, -burg*, English suffixes like *-town, -city* or Russian suffixes like *-grad*.

5.1.2 Word Context Features

Morphological information (POS and lemma) of three previous and one following words of the target word are used as features. The NE annotations of three previous tokens concatenated in a string is also considered as a feature of the Word Context Class. This feature has been seen as a dynamic one in the experiment. That means it depends on the previous decisions of the classifier. Another new informative 'in bracket'-feature looks whether the current token is in apostrophes.

5.1.3 Encoded Context (Word-Shapes)

These features carry information about the local context. The current token and its immediate context are encoded according to their orthographic pattern, which is derived equally for all tokens. In such a way, distinctive types of entities can be better detected, like web and email addresses (e.g. www.cip.ifi.lmu.com → xxx.xxx, email@gmx.de → xxx@x.xx), companies (e.g. GmbH → XxxX) and other organizations or proper names (e.g. EUROPARLAMENT → XXXXX).

5.1.4 Key-Words

Specific lists of key-words signal the belonging of a token to a particular NE category. For example, such words like *'denken', 'sagen'* may

[9] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger

	Strict			Loose			Outer			Inner		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Morphological	49.63	46.64	48.09	50.22	47.19	48.66	49.53	49.25	49.39	54.72	13.68	21.89
+ Context	75.18	61.09	67.41	75.78	61.57	67.94	76.06	63.20	69.04	59.35	34.43	43.58
+ Word-Shapes	76.15	65.59	70.48	76.83	66.18	71.11	77.22	67.69	72.14	58.45	39.15	46.89
+ Key-words	76.45	65.70	70.67	77.14	66.29	71.30	77.48	67.80	72.32	59.29	39.15	**47.16**
+ Gazetteers	76.76	65.94	**70.94**	77.45	66.53	**71.58**	77.84	68.06	**72.63**	58.87	39.15	47.03

Table 2: Results on the development set.

indicate PER NE; *'gründen', 'arbeiten'* are particular for ORG but also for PER; words *'Kino', 'Musik', 'Werk'* characterize the category other.

5.1.5 Gazetteers

Various gazetteers from different sources such as Wikipedia, DBpedia, the GeoNames geographical database etc. have been analysed. NEs were automatically extracted from these resources, categorized into different NE classes and written into lists. The size of the elaborated lists varies from 434 for category OTH to 339392 for category PER.

5.2 Inner Span

For the inner classifier a similar set of features has been used. However, the feature class *key-words* and the 'in bracket'-features are excluded as they lose their relevance for the sub-structure. The features from class *Word-Shapes* are also limited to two tokens.

Because the inner classifier is trained after the outer classifier, information about the NE tags the outer classifier assigns to the target, previous and following tokens is accessible. We use this information as additional features for the inner span.

Additionally, we include the NE tags of the three previous tokens for the inner span as a concatenated string.

6 Evaluation

DRIM has been evaluated on the development set provided by GermEval via the distributed scorer, which requires six tab-separated columns: index, token, first-level NEs (gold), second-level NEs (gold), first-level NEs (prediction), second-level NEs (prediction).

In our system, we define the baseline model where the NE tag probabilities depend on the morphological features with a current token, POS and lemma information, specifying token suffix, prefix and root. With these features, the system achieves an F-score of 48.09% (see first line of Table 2).

Including the features of the Word-Context-Class demonstrates that the performance of the NER system can be improved up to 19% (see second line of Table 2). Whereas, in other languages such morphological characteristics as capitalization are useful, for German it is almost impossible to find out the right definition of the word without a context. That is why using the information about POS, lemma and NE annotations of the surrounding words of the target token increases significantly the recognition of NEs in German.

Another important feature class is Word-Shapes. Using these features additionally to Morphological features and Word-Context features improved the F-score to 70.48% (see third line of Table 2).

Light improvements could be seen by adding Key-Words and Gazetteer features. With the Key-Words features the score is improved to 70.67% (see forth line of Table 2). We assume that Key-Word features would be better represented with the elaboration of the key words, particular to a certain category. Adding the Gazetteers features improves the final score to 70.94% (see fifth line of Table 2).

7 Conclusion and Future Work

The current work presented the SVM-based named entity recognition system DRIM and its participation at the GermEval 2014 NER Shared Task. The context of the current token has turned out to be the most informative feature class for NER for German. Experimental results on the strict (fine-grained) setting have shown a reasonably good system performance reaching 70.94% on the development set, and 69.33% on the final test set. In the future, we plan to explore variations of the current features, extending the Gazetteers and separating the common key words into groups particular to the different NE cate-

gories. Since context features have shown to be highly informative for this task, we plan on exploring further the optimal size of the context window that should be considered.

References

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. Nosta-d named entity annotation for german: Guidelines and dataset. In *Proceedings of LREC-14*.

Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. 1992. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pages 144–152, New York, NY, USA. ACM.

Asif Ekbal and Sivaji Bandyopadhyay. 2008. Bengali named entity recognition using support vector machine. In *Proceedings of Workshop on NER for South and South East Asian Languages, 3rd International Joint Conference on Natural Languge Processing (IJCNLP)*, pages 51–58, India.

Manaal Faruqui and Sebastian Padó. 2010. Training and evaluating a German named entity recognizer with semantic generalization. In *Proceedings of KONVENS 2010*, Saarbrücken, Germany.

Alexander Geyken and Norbert Schrader. 2006. LexikoNet, a lexical database based on role and type hierarchies. In *Proceedings of LREC*.

T. Kudo and Y. Matsumoto. 2000. Use of Support Vector Learning for Chunk Identification. In *Proceedings of Sixth Conference on Computational Natural Language Learning (CoNLL-2000)*.

Taku Kudo and Yuji Matsumoto. 2001. Chunking with support vector machines. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies*, NAACL '01, pages 1–8, Stroudsburg, PA, USA. Association for Computational Linguistics.

Andrei Mikheev, Claire Grover, and Marc Moens. 1998. Description of the ltg system used for muc-7. In *Proceedings of 7th Message Understanding Conference (MUC-7)*.

Marc Rössler. 2004. Corpus-based learning of lexical resources for german named entity recognition. In *LREC*. European Language Resources Association.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

Helmut Schmid. 1995. Improvements in part-of-speech tagging with an application to german. In *Proceedings of the ACL SIGDAT-Workshop*, pages 47–50.

Dan Shen, Jie Zhang, Guodong Zhou, Jian Su, and Chew-Lim Tan. 2003. Effective adaptation of hidden markov model-based named entity recognizer for biomedical domain. In *Proceedings of the ACL 2003 Workshop on Natural Language Processing in Biomedicine*, pages 49–56, Sapporo, Japan, July. Association for Computational Linguistics.

Mark Stevenson and Robert Gaizauskas. 2000. Using corpus-derived name lists for named entity recognition. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, ANLC '00, pages 290–295, Stroudsburg, PA, USA. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada.

Martin Volk and Simon Clematide. 2001. Learn - filter - apply - forget. mixed approaches to named entity recognition. In *Proceedings of the 6th International Workshop on Applications of Natural Language to Information Systems*, NLDB'01, pages 153–163. GI.

BECREATIVE: Annotation of German Named Entities

Fabian Dreer, Eduard Saller, Patrick Elsässer, Ulrike Handelshauser, Desislava Zhekova
CIS, Ludwig Maximilian University, Munich
{dreer|sallere|elsaesser|handelshauser}@cip.ifi.lmu.de
zhekova@cis.uni-muenchen.de

Abstract

This paper presents the BECREATIVE Named Entity Recognition system and its participation at the GermEval 2014 Named Entity Recognition Shared Task (Benikova et al., 2014a). BECREATIVE uses a hybrid approach of two commonly used procedural methods, namely list-based lookups and machine learning (Naive Bayes Classification), which centers around the classifier. BECREATIVE currently reaches an F-score of 37.34 on the strict evaluation setting applied on the development set provided by GermEval.

1 Introduction

Named Entity Recognition (NER) is an important part of many natural language processing (NLP) tasks first and foremost Information Extraction (IE), but as well necessary for question-answering systems and machine translation. In general, named entities (NEs) are phrases that represent persons, organizations, locations, times, quantities, etc. (Tjong Kim Sang and De Meulder, 2003). NER is the task of locating those phrases, mostly proper names, in an unstructured text and clustering them into a predefined set of categories.

The rest of this paper is organized as follows: In section 2, related work on the topic of NER that has been carried out over the last years is presented and discussed. Following, (in section 3) we shortly present the GermEval 2014 Shared Task (Benikova et al., 2014a) in the context of which the system was developed and evaluated. The description of BECREATIVE can be then found in section 4 that is followed by its evaluation (see section 5) and conclusion (section 6).

2 Related Work

Nowadays NER has reached numerous traditional domains, such as medicine or biology, but as well a more novel domain: The internet with all its blogs and social platforms where NER tools need to be less domain specific and thus perform quite differently than on an e.g. journalistic corpus. NER was first looked into more concretely back in 1990 (Nadeau and Sekine, 2007), when the main approaches were still based on heuristics and handcrafted rules. Shortly afterwards, it was already recognized as an essential subtasks of IE. The initial purpose was to extract structured information like names of persons, locations, organizations and also numeric values like time or date from newspaper articles or specialist literature. In 1995 at MUC-6 (Grishman and Sundheim, 1996) NER was constituted to be the initial goal for the first time, so "Named Entity" became an internationally accepted term in the world of natural language processing. Prerequisite for precise NER is the segmentation of data, performed by tokenization and chunking; for example "University of Munich" is a single NE, and the token "Munich" inside its span is also a NE. Yet, detecting all NEs (Carreras et al., 2002) and classifying them by their type still is a very challenging

task (Tjong Kim Sang and De Meulder, 2003). Besides NER on English texts, which is generally the language concentraiting most efforts, a small number of approaches for other languages were also carried out, such as (IREX) (Sekine and Isahara, 2000) for Japanese or as well the systems on German, Dutch or Spanish presented during the CoNLL 2002 and 2003 Shared Tasks on Language-Independent Named Entity Recognition (Tjong Kim Sang, 2002; Tjong Kim Sang and De Meulder, 2003). In the IREX project and the MUC-6 NE task (muc, 1995), new categories, such as *artifact*, *geographical* and *political entity* were added. Widening the NE types to a hierarchy containing more than 200 types and subtypes (Sekine et al., 2002) enabled new perspectives for Question Answering systems and NER on data from social media like twitter (Ritter et al., 2011). NER systems may use grammar-based techniques as well as statistical models like machine learning. Systems using handcrafted rules obtain better precision by the price of lower recall and extensive linguistic work. Statistic systems require a large amount of expensive manually annotated training data. Recently, hybrid approaches were also explored to sidestep the drawbacks of both main techniques (Nothman et al., 2013). Often, gazetteer-based NER systems are also developed or integrated within already existing approaches (Jahangir et al., 2012). Current NER technologies still lack in performance in specific domains, such as politics, molecular biology or yellow press. For both rule-based and statistic systems, opportunities for new solutions are created (Poibeau and Kosseim, 2001). Furthermore, the identification of relevant expressions in text and automatically linking them to Wikipedia is part of the recent scope of NLP challenges (Mihalcea and Csomai, 2007). Additionally should be noted that NER systems for German are not easily available or are closed source.

3 Task Description

The main aim of the GermEval 2014 Named Entity Recognition Shared Task (Benikova et al., 2014a) is not only the detection of NEs, but as well the extension of the task specifically to one language – German. Additionally, GermEval increases the level of NE embedding, also targeting

#	http://de.wikipedia.org/wiki/Manfred_Korfma		
1	Aufgrund	O	O
2	seiner	O	O
3	Initiative	O	O
4	fand	O	O
5	2001/2002	O	O
6	in	O	O
7	Stuttgart	B-LOC	O
8	,	O	O
9	Braunschweig	B-LOC	O
10	und	O	O
11	Bonn	B-LOC	O
12	eine	O	O
13	große	O	O
14	und	O	O
15	publizistisch	O	O
16	vielbeachtete	O	O
17	Troia-Ausstellung	B-LOCpart	O
18	statt	O	O
19	,	O	O
20	,,	O	O
21	Troia	B-OTH	B-LOC
22	-	I-OTH	O
23	Traum	I-OTH	O
24	und	I-OTH	O
25	Wirklichkeit	I-OTH	O
26	”	O	O
27	.	O	O

Figure 1: An example sentence of the GermEval data annotation format (Benikova et al., 2014b).

the identification of NEs inside already existing ones. Another peculiarity about the task is the fact that there are no restrictions regarding the types of NER systems as well as type and amount of used resources allowed for submission.

The data sets provided by the task consist mainly of articles extracted from the German Wikipedia and other News Corpora with over 31.000 sentences containing over 590.000 tokens. A sample of the data format can be seen in figure 1 (Benikova et al., 2014b). As the authors describe, the data is marked in the traditional BIO tagging scheme (Tjong Kim Sang and De Meulder, 2003) for the four main types: *person* (PER), *location* (LOC), *organization* (ORG) and *other* (OTH). Additionally, two subtypes with respect to all main classes are included: *part* and *deriv* indicating NE spans where only a subspan corresponds to a NE of the main types and respectively derivatives where the span is a derivation of a NE.

http://de.wikipedia.org

4 BECREATIVE

BECREATIVE is a Python implementation that makes use of the natural language toolkit (NLTK) that provides easy string handling, regular expression support and short development time. The current section provides further details about the system pipeline starting with preprocessing (see section 4.1), classification model (presented in section 4.2) and postprocessing (see section 4.3).

4.1 Preprocessing

During preprocessing, we bring the provided data, which is in a tab-separated value form, in a format that is better suited for our purpose. Internally we created a class representation for tokens, that mirrors the format of one row in the provided files and some empty fields for the tagger output, and one for sentences which is basically a list of tokens with some handy methods in addition. During the import, the data is already transformed to our representation of it, afterwards the data is annotated for part-of-speech (POS) by the TreeTagger developed by Helmut Schmid (Schmid, 1994; Schmid, 1995).

4.2 Naive Bayesian Classification

For NER proper, we train a Naive Bayesian classifier. The feature set used by the learner is presented in table 1. All feature representations are boolean values and the default weighting by the classifier is kept. The first 15 features are self-explanatory. Feature 16 checks if the second preceding token is a known NE (based on gazeteer lists collected from various online resources) and compares the preceding token against a list of verbs that indicate that the token could be a name. Feature 19 works similarly. Feature 17 checks the token for parts like *GmbH* or *Holding*, similar to 18 which tests for certain suffixes like *-hausen* or *ingen*. Feature 20 tests the second preceding token against a list of verbs, such as *wohnen* or *kommen* and looks the preceding token up in a list of prepositions.

4.3 Postprocessing

During postprocessing, gazetteer-based checks were additionally performed, which indicate a high probability of a token being a full or only part of a NE. The gazetteers were accumulated as lists for the following topics: Countries, Mountains, Waterbodies, Places of Interest, Street Names, Automobile Manufacturers, Book Titles, Film Titles, Styles, Forms of Address, First Names, Actors and Famous Persons.

As a final step, there is one list that contains phrases which are sure not to be Named Entities like measurements, so we are able to reduce the false positives a little further.

#	Description
1	The token itself
2	The preceding token
3	The following token
4	The token's index
5	The token's POS tag
6	The token's lemma
7	Capitalisation of the first letter
8	Capitalisation of the preceding word's first letter
9	Capitalisation of the following word's first letter
10	Whether the token matches a regular expression for a URL
11	Whether the token matches a regular expression for an IP address
12	Whether the token matches a regular expression for an email
13	Whether the token contains non letter characters
14	Whether the token contains numbers
15	Whether the token contains Roman numerals
16	Whether the token contextually could be a name
17	Whether the token has typical parts of an organization name
18	Whether the token has a location suffix
19	Whether the token contextually could be a location
20	Whether the token is one of certain verbs that stands usually with locations

Table 1: The feature set used by BECREATIVE

5 Evaluation

BECREATIVE was evaluated on the development set of the GermEval 2014 shared task. The results that the system achieves are presented in table 2. We also tested different subsets of the feature set. The first subset (*base*) includes features 1,2,3,4,7,8 and 9 from table 1, while the second subset (*base+POS*) adds the POS-tagger based features 5 and 6 as well. The performance of the full feature set is then listed under *all* in table 2. Additionally, after classification, the output of the classifier is also revised by our postprocessing gazetteer-based rules. leading to the system performance listed under *all+Lists* in table 2.

It is interesting to see (when the strict evaluation setting is observed) that including POS and lemma information in the feature set leads to a considerable decrease in system performance

http://www.python.org
http://www.nltk.org

setting	strict				loose				outer				inner			
	Acc.	**P**	**R**	**F1**	**Acc.**	**P**	**R**	**F1**	**Acc.**	**P**	**R**	**F1**	**Acc.**	**P**	**R**	**F1**
base	95.94	39.60	27.68	32.58	95.97	40.35	28.20	33.20	92.46	39.60	29.88	34.06	99.42	0.00	0.00	0.00
base+POS	92.75	18.07	42.21	25.31	92.83	19.06	44.51	26.69	86.07	18.07	45.58	25.88	99.42	0.00	0.00	0.00
all	95.90	38.66	31.89	34.95	95.93	39.29	32.42	35.52	92.38	38.66	34.44	36.43	99.42	0.00	0.00	0.00
all+Lists	95.97	39.58	35.34	37.34	95.99	40.20	35.90	37.93	92.51	39.58	38.16	38.86	99.42	0.00	0.00	0.00

Table 2: Results achieved by the BECREATIVE system based on the GermEval development set.

(from 32.58% for setting *base* to 25.31% for setting *base+POS*). This is due to the large decrease in precision (from 39.60% to 18.07%) even though recall is significantly improved (from 27.68% to 42.21%).

The combination of all features from table 2 leads to a system performance of 34.95% (see setting all), which is considerably low for a classification approach in comparison to state-of-the-art systems for German reported at the CoNLL-2003 Shared Task (Tjong Kim Sang and De Meulder, 2003).

Based on the setting for which all features are used ($all + Lists$), the detailed per class results given in table 3 show that BECREATIVE fails to identify most of the *part* and *deriv* subclasses (apart from LOCderiv and ORGpart). Additionally, all inner spans are also completely ignored by the system, which also contributes significantly to the overall low performance scores. This can be further approached by training two separate classifiers for both NE spans (outer and inner) and including span-specific or span-indicative features in both separate feature groups (e.g. classification decisions of the outer span can be included in the features for the inner span). Moreover, a task as NER would profit even more from sequential models (e.g. Conditional Random Fields) independent of the level of embedded phrases.

			P	**R**	**F1**
LOC	LOC	Outer strict	40.25	63.46	49.26
		Inner strict	0.00	0.00	0.00
		Outer loose	42.78	52.69	47.22
		Inner loose	0.00	0.00	0.00
	LOCderiv	Outer strict	63.95	23.91	34.81
		Inner strict	0.00	0.00	0.00
	LOCpart	Outer strict	0.00	0.00	0.00
		Inner strict	0.00	0.00	0.00
ORG	ORG	Outer strict	27.21	25.45	26.30
		Inner strict	0.00	0.00	0.00
		Outer loose	28.24	22.62	25.12
		Inner loose	0.00	0.00	0.00
	ORGderiv	Outer strict	0.00	0.00	0.00
		Inner strict	0.00	0.00	0.00
	ORGpart	Outer strict	37.50	3.30	6.06
		Inner strict	0.00	0.00	0.00
OTH	OTH	Outer strict	51.24	22.96	31.71
		Inner strict	0.00	0.00	0.00
		Outer loose	51.24	20.46	29.25
		Inner loose	0.00	0.00	0.00
	OTHderiv	Outer strict	0.00	0.00	0.00
		Inner strict	0.00	0.00	0.00
	OTHpart	Outer strict	0.00	0.00	0.00
		Inner strict	0.00	0.00	0.00
PER	PER	Outer strict	41.65	40.65	41.15
		Inner strict	0.00	0.00	0.00
		Outer loose	41.65	39.53	40.57
		Inner loose	0.00	0.00	0.00
	PERderiv	Outer strict	0.00	0.00	0.00
		Inner strict	0.00	0.00	0.00
	PERpart	Outer strict	0.00	0.00	0.00
		Inner strict	0.00	0.00	0.00

Table 3: Results per class achieved by BECREATIVE based on the GermEval development set.

6 Future Work and Conclusion

The current paper presented the BECREATIVE system for NER developed and evaluated in the context of the GermEval 2014 Named Entity Recognition Shared Task. BECREATIVE combines a Naive Bayesian Classifier with rules performing gazetteer-based checkup and achieves a performance of 37.34 on the development set.

In the future, we plan to explore further features (e.g. investigating for example a larger context than just preceeding and following tokens) for the classification approach in order to improve the still considerably low learner performance. Additionally, as noted above, we would also like to apply sequential models to the task and include a separate classification for each layer of embedding present in the data.

References

Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Pado. 2014a. Germeval 2014 named entity recognition: Companion paper. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014b. NoSta-D Named Entity Annotation for German: Guidelines and Dataset. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Xavier Carreras, Lluís Màrquez, and Lluís Padró. 2002. Named entity extraction using adaboost. In *Proceedings of the 6th Conference on Natural Language Learning - Volume 20*, COLING-02, pages 1–4, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ralph Grishman and Beth Sundheim. 1996. Message understanding conference-6: A brief history. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*, COLING '96, pages 466–471, Stroudsburg, PA, USA. Association for Computational Linguistics.

Faryal Jahangir, Anwar Waqas, Bajwa Usama Ijaz, and Wang Xuan. 2012. N-gram and gazetteer list based named entity recognition for urdu: A scarce resourced language. In *Proceedings of the 10th Workshop on Asian Language Resources*, page 95–104.

Rada Mihalcea and Andras Csomai. 2007. Wikify!: linking documents to encyclopedic knowledge. In *In CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 233–242. ACM.

1995. *MUC6 '95: Proceedings of the 6th Conference on Message Understanding*, Stroudsburg, PA, USA. Association for Computational Linguistics.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R. Curran. 2013. Learning multilingual named entity recognition from wikipedia. *Artif. Intell.*, 194:151–175.

Thierry Poibeau and Leila Kosseim. 2001. Proper name extraction from non-journalistic texts. In *In Computational Linguistics in the Netherlands*, pages 144–157.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: an experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1524–1534, Stroudsburg, PA, USA. Association for Computational Linguistics.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

Helmut Schmid. 1995. Improvements In Part-of-Speech Tagging With an Application to German. In *Proceedings of the ACL SIGDAT-Workshop*, pages 47–50, Dublin, Ireland.

Satoshi Sekine and Hitoshi Isahara. 2000. Irex: Ir and ie evaluation project in japanese. In *Proceedings of International Conference on Language Resources & Evaluation (LREC 2000)*.

Satoshi Sekine, Kiyoshi Sudo, and Chikashi Nobata. 2002. Extended named entity hierarchy.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada.

Erik F. Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 Shared Task: Language-independent Named Entity Recognition. In *Proceedings of the 6th Conference on Natural Language Learning - Volume 20*, COLING-02, pages 1–4, Stroudsburg, PA, USA. Association for Computational Linguistics.

Nessy: A Hybrid Approach to Named Entity Recognition for German

Martin Hermann, Michael Hochleitner, Sarah Kellner, Simon Preissner, Desislava Zhekova
CIS, Ludwig Maximilian University, Munich
{Martin.Hermann, M.Hochleitner, S.Kellner, Simon.Preissner, D.Zhekova}
@campus.lmu.de

Abstract

In this paper we present Nessy (Named Entity Searching System) and its application to German in the context of the GermEval 2014 Named Entity Recognition Shared Task (Benikova et al., 2014a). We tackle the challenge by using a combination of machine learning (Naive Bayes classification) and rule-based methods. Altogether, Nessy achieves an F-score of 58.78% on the final test set.

1 Introduction

Named Entity Recognition (NER) is a subtask of information extraction and is an important topic in natural language processing. It is useful for the identification of where information is located, how it may be connected and used for tasks such as text classification (Gui et al., 2012) and question answering (Mollá et al., 2006).

However, NER is not a simple task, especially for German, where capitalization is not as informative as in many other languages, such as English or Spanish. Following the NE annotation guidelines presented by Benikova et al. (2014b), the GermEval Shared Task on Named Entity Recognition (Benikova et al., 2014a) aims at detecting named entities (NEs) and assigning them to one of four classes: persons (-PER), locations (-LOC), organizations (-ORG), and the class of other (-OTH), where those NEs are assigned to which cannot be matched with the aforementioned classes. Furthermore, there are two subclasses (-part and -deriv) which are used for NEs that are subparts of bigger entities (-part, e.g. *deutschlandweit*) or derivatives (-deriv, e.g. *Bremer* Staatsanwaltschaft).

Named Entity Recognition and Classification (NERC) was introduced as a subtask of Information Extraction (IE) at the 6th Message Understanding Conference (MUC-6) in 1995 (Nadeau and Sekine, 2007). Since then, remarkable results have been reached for NER in English. Systems at the 7th Message Understanding Conference (MUC-7) reached scores of up to 93% (Mikheev et al., 1998), which is close to the inter-annotator agreement 96% for that task (Chinchor, 1998). So far, most work in NER for German was conducted in the context of the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition (Tjong Kim Sang and De Meulder, 2003). The systems reached F-scores of 72.41% on the German test set and 88.76% on the English test set. Among the machine learning techniques used for CoNLL-2003 Maximum Entropy (MaxEnt) and Hidden Markov Models (HMM) were most popular (Tjong Kim Sang and De Meulder, 2003).

Combining different classifiers also proved to be beneficial. Florian et al. (2003), for example, added robust linear classifier and transformation-based learning to MaxEnt and HMM. Additionally, to improve the performance of classification, it was common to make use of gazetteers.

Unfortunately, for German, there are not many freely available and simultaneously high-

fgroup	name	description
d_lex	*pos*	POS-tag of the token
	word	token itself
d_other	*prev_dec*	preceding IOB-tag
	all_caps	check if all characters are uppercased

Table 1: The feature groups (fgroup) used for NED.

performance NERs. One such system that applies semantic generalizations learned from unlabelled data was presented by Faruqui and Padó (2010).

In this paper, we describe the NER system Nessy developed for the GermEval 2014 Shared Task. We break NER down into two steps: named entity detection and named entity classification, both described in section 2 where all further details about the system pipeline are presented. In section 3, we provide a discussion on the results achieved by Nessy on the development set provided by the GermEval 2014 Shared Task and in section 4 we conclude our work.

2 The Nessy System

2.1 Preprocessing

Part-of-Speech (POS) tags and lemmas were acquired via the TreeTagger (Schmid, 1994; Schmid, 1995). Additionally large lists of known NEs (gazetteers) were prepared (containing 68922 entries). These NEs were taken directly from the already manually annotated data provided by the CoNLL-2003.

2.2 Named Entity Detection

For the task of named entity detection (NED), we use a Naive Bayes classifier and tag each of the words in an IOB-manner. The small set of features currently used in this classifier are presented in table 1. To make sure that the output contains only valid IOB-sequences any isolated I-tag is converted into a B-tag.

2.3 Named Entity Classification

For Named Entity Classification (NEC), we extract the presumable named entities found during NED. Again, these are passed to a naive Bayes classifier that uses the features given in table 2. In the case of one-word-entities, the features *ne*, *first_t* and *last_t* contain the same information. The feature *in_lookup* checks against the gazetteers prepared during preprocessing.

fgroup	name	description
c_lex	*ne*	the named entity itself
	lemmas	the sequence of lemmas in the NE
	first_t	the first word of the respective NE
	last_t	the last word of the respective NE
c_cont	*prev_t*	the word preceding the NE
	foll_t	the word following the NE
c_other	*num_t*	number of tokens in the NE
	all_caps	check if all characters are uppercased
	in_lookup	gazetteer lookup

Table 2: The feature groups (fgroup) used for NEC.

2.4 "part" and "deriv" Subclasses

Tags labeled with "part" and "deriv" are an individual characteristic of this data. Although many of them are already correctly found by the classifier, additional steps proved to be necessary.

2.4.1 The "part" Subclass

Tags ending in "part" are used to annotate tokens that are not NEs themselves, but contain a substring that does qualify as such. They make up about 5.5% of NEs in the training and 6.4% in the development data, most of which (96.4% in the training, 97.3% in the development data) occur in the outer layer. Hence, we neglect the inner layer completely in this step. Additionally, as we simply "overwrite" previously assigned tags, this may also correct mistakes in the detection step (e.g., if the phrase *EU-Kommissarin Viviane Reding* is (incorrectly) marked with "PER", detection of *EU-Kommissarin* as "ORGpart" would not only label this token appropriately, but also correct the span of *Viviane Reding*. Had we written the "ORGpart" label in the inner layer, we would end up with two wrong annotations.)

The detection of "part" tags is done with four lists of single-word NEs, one for every category, compiled from the training data and expanded with the list of stems described below. The list is revised, such that only entries are allowed that occur more often as a NE of the given category than not, in order to reduce ambiguity that may arise from either inaccuracies in the data, or, more likely, language itself (e.g. many surnames, such as *Gold*, are also common nouns).

By far, the biggest part (77.9% in the training, 77.7% in the development data) of partial NEs contains one or more hyphens ("-"), and in turn, a considerable amount of tokens (19.8% in the

NEs that are missing their "B-" tag are corrected.

training, 22.7% in the development data) containing hyphens are labeled with the "part" subclass, so it seems sensible to focus on these. Such tokens are separated at the hyphens and the first part is checked against the lists of single-word NEs. If a match is found, the token is labeled accordingly.

2.4.2 The "deriv" Subclass

Derivated forms of NEs are marked with tags ending in "deriv". As they account for about 11.9% of NE in the training and 10.5% in the development data, they should not be neglected. Especially LOCderiv, such as *deutschen* (German) or *Engländer* (Englishman) are very common in all datasets. Unlike the "part" labels, a considerable amount (16.5% in the training, 15.8% in the development data) of tags with "deriv" is found in the inner layer, so it is more reasonable here to check if the derivated form may already be part of a larger NE.

Similar to the "part" labels, we use four lists of single-word candidates, although this time, the entries are not simply taken from the training data, but suitable entries found there are stemmed, and then the stems are combined with a list of possible endings, e.g. *-lich*, *-istischer* or *-erin*. However, controlling this list with the test data is even more important than in the previous case, as from *deut*, which is generated as stem of *deutsch* (albeit linguistically not entirely correct) not only *deutsches*, *deutscher* or *deutsche* are derived, but also *deutlich* (clearly) or *deutung* (interpretation), which would cause many false-positives. A lot of nonsensical words are also generated, such as **deutistisch*, but as they seldom appear, they do not need to be considered.

2.5 Inner Layer

The data contains recursive NEs to the depth of one nested layer. This inner layer is filled with some of the "deriv" labeled tags and some NE found in the postprocessing step, but it is reasonable to further search for possible nested NEs. As they can only occur if the outer layer is not empty, the search is done only within previously found NEs. Here, we make further use of the list of NEs that has been compiled for finding "part" tags, as it proves to yield better results at this point than the gazetteers compiled from the CoNLL-2003 data

2.6 Additional Rules

Several rules have been written that account for special cases of NEs. These can be grouped into four different classes:

Hyperlinks: Hyperlinks are always annotated as NEs of the category OTH.

Hyphens: While hyphens usually are a sign for the "part" subclass (as described above), compounds that contain one or more hyphens and end in a NE usually obtain the class of that NE. This is so, since in German the last part of a word determines its class. So, for example, while both *Taiwan* and *Dollar* in *Taiwan-Dollar* are NEs, *Taiwan-Dollar* is a form of *Dollar*, and therefore should be categorized as OTH, just like *Dollar* itself.

Split-off parts: A hyphen at the end of a token (e.g. *Süd-*) and tokens such as *und* (and) or *oder* (or) following it may indicate split-off parts (e.g. *Süd- und Nordkorea*), both of which should have the class of the second token, in this case, LOC.

Tokens following nationalities: Nessy tends to mark any nationality and its following token as a two-word-NE. This, however, is hardly ever the case, unless the nationality starts with an uppercase letter (e.g. *Deutsches Theater*). Such subsequent tokens are discarded by using a list of nationalities during postprocessing.

3 Evaluation

The Nessy system was evaluated on the development set provided by GermEval 2014 (Benikova et al., 2014a). The results on the development and final test set are given in table 3. In order to see how informative the different feature types are (given in table 2), we evaluate separately a number of forward/backward inclusion/exclusion settings on the development data. First, we test each of the different feature groups separately, leading to settings +c_cont, +c_lex, +c_other in table 3 and then, we report results by excluding one of the groups, leading to settings -c_cont, -c_lex, -c_other. All three groups together are marked as +all in the table. Additionally, all seven variations are once tested on their own (-R) and once

Cases such as *EU-*, where the only hyphen in the word is at the end, are checked against.

setting	Metric 1 (Strict)				Metric 2 (Loose)				Metric 3 - Outer Chunks				Metric 3 - Inner Chunks			
	Acc.	**P**	**R**	**F1**	**Acc.**	**P**	**R**	**F1**	**Acc.**	**P**	**R**	**F1**	**Acc.**	**P**	**R**	**F1**
+c_cont-R	95.99	42.46	39.54	40.95	96.08	44.03	40.99	42.45	92.94	45.32	40.73	42.90	99.04	18.31	24.53	20.97
+c_other-R	96.16	47.79	44.14	45.89	96.24	49.89	46.08	47.91	93.20	50.33	45.21	47.64	99.13	24.62	30.66	27.31
+c_lex-R	96.74	55.37	47.89	51.36	96.83	57.25	49.51	53.10	94.02	55.44	49.89	52.52	99.45	53.33	22.64	31.79
-c_lex-R	96.59	55.26	50.42	52.73	96.64	56.29	51.35	53.71	93.96	57.81	51.91	54.70	99.22	28.88	31.60	30.18
-c_cont-R	96.77	59.02	52.81	55.74	96.81	59.91	53.60	56.58	94.23	60.82	54.67	57.58	99.31	34.83	29.25	31.79
-c_other-R	96.93	60.54	53.22	56.65	96.98	61.37	53.95	57.42	94.47	61.62	55.35	58.31	99.39	41.48	26.42	32.28
+all-R	96.90	61.40	55.06	58.06	96.94	62.13	55.72	58.75	94.49	63.56	57.07	60.14	99.30	33.69	29.72	31.58
+c_cont+R	96.13	44.00	40.68	42.28	96.21	45.58	42.13	43.79	93.12	46.19	41.92	43.95	99.15	21.99	25.00	23.40
+c_other+R	96.33	50.02	45.77	47.80	96.42	52.56	48.09	50.23	93.42	51.73	46.93	49.22	99.25	30.70	31.13	30.91
+c_lex+R	96.82	57.35	50.00	53.42	96.91	59.14	51.56	55.09	94.17	57.39	52.13	54.63	99.46	56.32	23.11	32.78
-c_lex+R	96.78	57.96	52.36	55.02	96.82	58.96	53.26	55.96	94.21	59.51	53.96	56.60	99.35	37.36	32.08	34.52
-c_cont+R	96.94	61.76	54.9	58.16	96.97	62.62	55.72	58.97	94.47	62.73	56.96	59.70	99.41	45.00	29.72	**35.80**
-c_other+R	97.02	62.43	55.27	58.63	97.07	63.25	55.99	59.40	94.62	63.40	57.52	60.31	99.41	44.19	26.89	33.43
+all+R	97.06	64.04	57.14	**60.39**	97.10	64.74	57.76	**61.05**	94.72	65.36	59.27	**62.17**	99.40	42.67	30.19	35.36
final test	97,07	63,57	54,65	58,78	97,11	64,34	55,31	59,48	94,77	64,83	56,93	60,62	99,38	42,86	27,38	33,41

Table 3: System results achieved on the GermEval 2014 development (upper part) and official test (last row) set.

with the supplementary use of the handcrafted rules presented in section 2.6, (+R).

As can be seen from the results of the strict evaluation setting (Metric 1), most informative to the learner on its own was the group of lexical features (c_lex), which reaches F-score of 51.36% when used alone during classification (setting *+c_lex-R*). This is a considerably big contribution regarding the fact that this feature group consists of four basic features representing the tokens and lemmas contained in one NE span. The other two groups (c_cont and c_other) also seem to carry very valuable information for the recognition process reaching scores of 40.95% and 45.89% respectively (settings *+c_cont-R* and *+c_other-R*), showing that both contextual and features carrying information about the number of tokens in a NE, their capitalization and presence in gazetteers should not be ignored for this task. The combination of all three groups (setting *+all-R*), reaches an improved F-score of 58.06%.

All these settings are then combined with the use of manually created rules leading to the *+R* settings in table 3. What can be seen is that the used rules do not interact with the separate feature group contribution, which leads to the same result tendencies as without the application of rules. However, the latter do increase the system performance for all tested variations, leading to an F-score of 60.39% (see setting *+all+R*), which is the highest score of our system based on the development set. Such a performance is competitive to the performance of systems applied to German on the CoNLL-2003 Shared Task ranging between F-scores of 47.74% to 72.41% (Tjong Kim Sang and De Meulder, 2003). We consider this to be a very good performance given the small feature set we employ.

The F-score of 60.39% is based mainly on the system performance for the outer layer of NE (62.17%), which seems to be weaker for the inner layer (achieving 35.36%). In fact, with respect to the inner layer, the system reaches best scores (35.80%) when context features are not used (setting *-c_cont+R*), which is surprising, since these features deliver information from the outer span, which should indicate the type of the outer NE in which the inner NE is included.

4 Future Work and Conclusion

In this paper, we presented the participation of Nessy, which is a hybrid approach to NER, at the GermEval 2014 Named Entity Recognition Shared Task for German. We evaluated the system (using Metric 1) on the development set provided by GermEval 2014, reaching an F-score of 60.39% on the development set and 58,78% on the final test set, which is considerably good for the small feature set that the system employs.

In the future, we would like to look deeper into the use of world knowledge for NER and explore the use of features carrying information about possible semantic relations between the tokens present in the NEs and tokens included in already known NEs present in available gazetteers.

References

[Benikova et al.2014a] Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Pado. 2014a. Germeval 2014 named entity recognition: Companion paper. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

[Benikova et al.2014b] Darina Benikova, Chris Biemann, and Marc Reznicek. 2014b. NoSta-D Named Entity Annotation for German: Guidelines and Dataset. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

[Chinchor1998] Nancy A. Chinchor. 1998. Proceedings of the Seventh Message Understanding Conference (MUC-7) Named Entity Task Definition. page 21 pages, Fairfax, VA.

[Faruqui and Padó2010] Manaal Faruqui and Sebastian Padó. 2010. Training and Evaluating a German Named Entity Recognizer with Semantic Generalization. In *Proceedings of KONVENS 2010*, page 129. Semantic Approaches in Natural Language Processing.

[Florian et al.2003] Radu Florian, Abe Ittycheriah, Hongyan Jing, and Tong Zhang. 2003. Named Entity Recognition Through Classifier Combination. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 168–171, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Gui et al.2012] Yaocheng Gui, Zhiqiang Gao, Renyong Li, and Xin Yang. 2012. Hierarchical Text Classification for News Articles Based-on Named Entities. In Shuigeng Zhou, Songmao Zhang, and George Karypis, editors, *ADMA*, volume 7713 of *Lecture Notes in Computer Science*, pages 318–329. Springer.

[Mikheev et al.1998] Andrei Mikheev, Claire Grover, and Marc Moens. 1998. Description of the LTG system used for MUC-7. In *In Proceedings of 7th Message Understanding Conference (MUC-7)*.

[Mollá et al.2006] Diego Mollá, Menno van Zaanen, and Daniel Smith. 2006. Named Entity Recognition for Question Answering. In *Proceedings of the 2006 Australasian Language Technology Workshop (ALTW2006)*, pages 51–58.

[Nadeau and Sekine2007] David Nadeau and Satoshi Sekine. 2007. A Survey of Named Entity Recognition and Classification. *Linguisticae Investigationes*, 30(1):3–26, January.

[Schmid1994] Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK.

[Schmid1995] Helmut Schmid. 1995. Improvements in Part-of-Speech Tagging with an Application to German. In *In Proceedings of the ACL SIGDAT-Workshop*, pages 47–50.

[Tjong Kim Sang and De Meulder2003] Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada.

Semi-Supervised Neural Networks for Nested Named Entity Recognition *

Jinseok Nam
Knowledge Engineering Group
Department of Computer Science, TU Darmstadt, Germany
Knowledge Discovery in Scientific Literature
German Institute for Educational Research, Germany
nam@cs.tu-darmstadt.de

Abstract

In this paper, we investigate a semi-supervised learning approach based on neural networks for nested named entity recognition on the GermEval 2014 dataset. The dataset consists of triples of a word, a named entity associated with that word in the first-level and one in the second-level. Additionally, the tag distribution is highly skewed, that is, the number of occurrences of certain types of tags is too small. Hence, we present a unified neural network architecture to deal with named entities in both levels simultaneously and to improve generalization performance on the classes that have a small number of labelled examples.

1 Introduction

Named Entity Recognition (NER) is an important natural language processing (NLP) task that aims at assigning a class label to a word such as person, location, organization and so on. In contrast to the traditional NER where a classifier assigns only a single named entity (NE) for elements in text, the GermEval 2014 dataset (Benikova et al., 2014b) allows for elements to have two NEs at most. For example, "TU Darmstadt" is not only considered as an *organization*, but "Darmstadt" can be also tagged as a *location*. The dataset consists of sentences sampled from Leipzig Corpora Collection (LCC) (Quasthoff et al., 2006) publicly available for download.[1]

[1] http://corpora.uni-leipzig.de/download.html

Recently, neural networks (NNs) have succeeded in various NLP tasks including NER (Collobert et al., 2011). Thus, we build a neural network architecture solving the nested NER problem in a semi-superivsed way by making use of a large number of unlabelled sentences from LCC.

2 Background

2.1 NER using Neural Networks

Collobert et al. (2011) proposed a unified neural network architecture, namely SENNA, on which we build an architecture for nested NER.

Consider a sentence $t = \{w_1, w_2, \cdots, w_{N_t}\}$ of length N_t in which each word w_i is associated with its target y_i, which has one of C possible tags. The inputs to SENNA are the concatenated vector representations for the words in the sentence. The vector representations can be drawn from a matrix $\mathbf{L} \in \mathbb{R}^{d \times |V|}$ where d is the dimension of the vectors and $|V|$ is the number of words in our vocabulary. While it is possible to define another feature matrix that we want to learn such as capitalization features $\mathbf{L}^{(caps)}$ as well as the word features $\mathbf{L}^{(w)}$, for simplicity, we only consider the word features as $\mathbf{L}$ in this Section.

Assuming that we wish to tag a word w_i and let k_w be the width of a window. The vector representations of word w_i and of words surrounding w_i in a window are drawn from $\mathbf{L}$, then concatenated to form $\mathbf{x}_i = \{\mathbf{L}_{\cdot w_n}\}_{n=-\lfloor k_w/2 \rfloor + i}^{\lfloor k_w/2 \rfloor + i} \in \mathbb{R}^{d \cdot k_w}$ where $\lfloor x \rfloor$ denotes the largest integer not greater than x. If n is less than 1 or greater than N_t, a special *padding* word is used instead. In turn, the input $\mathbf{x}_i$ is passed to a non-linear function to obtain a hidden representation

$$\mathbf{h}_i = f\left(\mathbf{W}^{(1)}\mathbf{x}_i + \mathbf{b}^{(1)}\right) \quad (1)$$

where the function $f : \mathbb{R} \rightarrow \mathbb{R}$ is an element-wise transfer function, e.g., *sigmoid*, *tanh*, and *ReLU*,

$\mathbf{W}^{(1)} \in \mathbb{R}^{F\times(d\cdot k_w)}$ is a matrix of weights that link input units to hidden units, and $\mathbf{b}^{(1)} \in \mathbb{R}^F$ is a vector of biases for the hidden layer. The hidden representation $\mathbf{h}_i$ is, then, fed forward to the output layer to yield the prediction scores $\hat{\mathbf{y}}_i \in \mathbb{R}^C$ of the tags for the given local context

$$\hat{\mathbf{y}}_i = \mathbf{W}^{(2)}\mathbf{h}_i + \mathbf{b}^{(2)} \quad (2)$$

where $\mathbf{W}^{(2)} \in \mathbb{R}^{C\times F}$ are the weights between hidden and output units, each of which corresponds to a tag, and $\mathbf{b}^{(2)} \in \mathbb{R}^C$ are the biases for the output layer.

If we assume that the tag of each word depends only on that word and its context, i.e., $\mathbf{x}_i$, then the probability distribution over $\{w_i, y_i\}$ can be formulated as follows

$$p\left(y_1, \cdots, y_{N_t}, w_1, \cdots, w_{N_t}\right) = \prod_{i=1}^{N_t} p\left(y_i|\mathbf{x}_i; \Theta\right). \quad (3)$$

In order to convert the prediction scores $\hat{y}_{ji}$ of the tag j for the word w_i into probability, we can use the *softmax* function

$$p\left(y_{ji} = 1|\mathbf{x}_i; \Theta\right) = \frac{\exp \hat{y}_{ji}}{\sum_k \exp \hat{y}_{ki}} \quad (4)$$

where $\Theta = \{\mathbf{L}, \mathbf{W}^{(1)}, \mathbf{b}^{(1)}\mathbf{W}^{(2)}, \mathbf{b}^{(2)}\}$ is a set of parameters. By taking the *log*, our objective, Eq. 4, becomes

$$\max_{\Theta} \sum_{i=1}^{N_t}\sum_{j=1}^{C} \mathbb{I}\left[y_{ji} = 1\right]\left(\hat{y}_{ji} - \log \sum_{k=1}^{C} \exp \hat{y}_{ki}\right) \quad (5)$$

where $\mathbb{I}[\cdot]$ denotes the indicator function which takes 1 when the argument is true, otherwise 0. This is referred to as *word-level* log-likelihood.

Learning Tag Dependencies In *word-level* log-likelihood, tag dependencies are ignored by the assumption that a tag is determined by only its local context. To exploit dependencies between tags, we take tag transition scores $\mathbf{T} \in \mathbb{R}^{C\times C}$ into account. A prediction score for the whole sentence is given by

$$\hat{y}_{[\mathbf{c}]} = \sum_{i=1}^{N_t} \mathbf{W}^{(2)}\mathbf{h}_i + \mathbf{b}^{(2)} + T_{c_i, c_{i-1}} \quad (6)$$

where $[\mathbf{c}]$ denotes a sequence of the tags in the sentence, c_i indicates the tag of the word w_i, and $T_{c_i,c_{i-1}}$ is a transition score from c_{i-1} to c_i. For the case $i = 1$, we also need initial tag scores $T_{c,0} \in \mathbb{R}^C$. The prediction score for the sentence is also transformed to a probability divided by the scores over all possible tag sequences $[\mathbf{k}]$

$$p\left(\{y_i\}_{i=1}^{N_t}|\{\mathbf{x}_i\}_{i=1}^{N_t}; \Theta, \mathbf{T}\right) = \frac{\exp \hat{y}_{[\mathbf{c}]}}{\sum_{[\mathbf{k}]} \exp \hat{y}_{[\mathbf{k}]}}. \quad (7)$$

Similarly, the objective taking transitions between tags into consideration is given by

$$\max_{\Theta, \mathbf{T}} \hat{y}_{[\mathbf{c}]} - \log \sum_{[\mathbf{k}]} \exp \hat{y}_{[\mathbf{k}]} \quad (8)$$

which is referred to as *sentence-level* log-likelihood and this can be addressed efficiently using *recursion*.

2.2 Semi-Supervised Learning

The simplest algorithm for semi-supervised learning is self-training (Rosenberg et al., 2005). In self-training, once a model is trained on labelled data, it is used to predict labels of unlabelled data, then such unlabelled data are provided as if additional labelled examples.

Pseudo-Label (PL) (Lee, 2013) is a semi-supervised learning technique especially for NNs. Unlike self-training, it estimates pseudo-labels, most probable labels of unlabelled data, during training and uses them to update parameters as well as labelled examples. Its purpose is similar to Entropy Regularization (Grandvalet and Bengio, 2005) that minimizes conditional entropy of unlabelled data as a measure of class overlap on the feature space.

3 Semi-Supervised Neural Networks for Nested NER

In contrast to the traditional NER, a word in nested NER can be tagged by multiple NEs. For simplicity, the number of levels is limited to two.

3.1 Jointly Learning Top-level and Nested NEs

In nested NER, a sentence t can be characterized by a sequence of triples $\{w_i, y_i^1, y_i^2\}$ where y_i^1 is the tag of the word w_i in the first level, and y_i^2 for the second level. Note that the tags in both levels are defined over the same set. Figure 1 describes our proposed architecture to tackle nested NER.

The proposed model deals with all NEs in both levels jointly during the learning phase by using an additional feature matrix $\mathbf{L}^{(ne)} \in \mathbb{R}^{d_{ne}\times C}$ for NEs, which is also a set of learnable parameters like $\mathbf{L}^{(w)}$. Each column of $\mathbf{L}^{(ne)}$ corre-

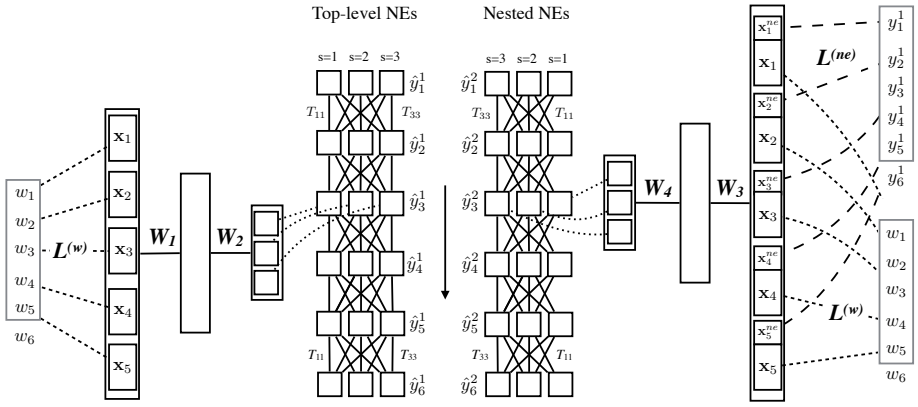


Figure 1: An illustrative example of the proposed architecture for jointly learning top-level and nested NEs. Consider a sentence $t = \{w_i, y_i^1, y_i^2\}_{i=1}^6$ of length 6, a window of size $k_w = 5$, and that we want to predict tags y_3^1, y_3^2 for a word w_3. Assuming that the number of NEs in the problem is 3, s indicates an index of a named entity. A matrix of word embeddings $\mathbf{L}^{(w)}$ and the tag transition matrix $\mathbf{T}$ are shared between two networks. Each network is trained to make predictions NEs of given a word sequence $\{w_i\}_{i=1}^6$ for each level.

sponds to a vector representation of a named entity. Given the concatenated feature vectors of a word $\mathbf{x}_i$ in the window as described in Section 2.1, we construct a vector representation for NEs in the top-level corresponding to that word, denoted by $\mathbf{x}_i^{ne} \in R^{d_{ne} \cdot k_w}$, then concatenate it to $\mathbf{x}_i$, which yields *combined* vector representations of words and NEs $\mathbf{x}_i^{comb} = \{\mathbf{x}_i, \mathbf{x}_i^{ne}\} \in R^{(d_{ne}+d_K)\times k_w}$. Similar to Eq.7 for the first level NEs, the *sentence-level* log-likelihood is also computed for the second level NEs like $\hat{\mathbf{y}}_i^2 = \mathbf{W}^{(4)} f\left(\mathbf{W}^{(3)}\mathbf{x}_i^{comb} + \mathbf{b}^{(3)}\right) + \mathbf{b}^{(4)}$. Then, the training objective considering the first- and second-level NEs simultaneously is given by

$$
\begin{aligned}
& p\left(\{y_i^1, y_i^2, w_i\}_{i=1}^{N_t}; \bar{\Theta}\right) = \\
& (1-\alpha)\; p\left(\{y_i^1\}_{i=1}^{N_t} | \{\mathbf{x}_i\}_{i=1}^{N_t}; \Theta, \mathbf{T}\right) \\
& + \alpha\, p\left(\{y_i^2\}_{i=1}^{N_t} | \{\mathbf{x}_i^{comb}\}_{i=1}^{N_t}; \theta, \mathbf{T}\right)
\end{aligned} \tag{9}
$$

where $\theta = \{\mathbf{W}^{(3)}, \mathbf{b}^{(3)}, \mathbf{W}^{(4)}, \mathbf{b}^{(4)}, \mathbf{L}^{(\cdot)}\}$, $\bar{\Theta} = \{\Theta, \theta, \mathbf{T}\}$, and $\alpha \in [0, 1]$ is a control parameter.

3.2 Learning from Pseudo Labels of Unlabelled Data

Semi-supervised learning methods are well-suited to the problems where the number of training instances is insufficient. Tag distribution of the GermEval dataset is highly skewed. In other words, the proportion of the three tag types, i.e., LOC, PER, and ORG, amounts to approximately 70% (See (Benikova et al., 2014b) for statistics).

In this work, we apply PL to only the first level in order to improve the generalization performance on such small classes. The first term of the right hand side in Eq. 9 can be re-written as

$$
\begin{aligned}
& (1-\alpha)\; p\left(\{y_i^1\}_{i=1}^{N_t} | \{\mathbf{x}_i\}_{i=1}^{N_t}; \Theta, \mathbf{T}\right) \\
& + (1-\alpha)\, \beta_t\, p\left(\{\hat{y}_{ui}^1\}_{ui=1}^{uN_t} | \{\mathbf{x}_{ui}\}_{ui=1}^{uN_t}; \Theta, \mathbf{T}\right)
\end{aligned} \tag{10}
$$

where *ui* is an index of an unlabelled sentence randomly selected from LCC, $\hat{y}_{ui}^1$ is a pseudo tag for the word w_{ui} in an un-annotated sentence, and β controls the importance of learning from unlabelled data. Scheduling the control parameter at time t takes the following form:

$$
\beta_t = \begin{cases} 0 & t < T_1 \\ \frac{t-T_1}{T_2-T_1}\beta_{\max} & T_1 \le t \le T_2 \\ \beta_{\max} & t > T_2 \end{cases} \tag{11}
$$

with $\beta_{\max} = 2$, $T_1 = 100$, and $T_2 = 500$. [2] The pseudo label $\hat{y}_{ui}^1$ is determined by simply choosing the most confident one given prediction scores for an un-annotated sentence during training.

4 Experiments

Our experiments were performed on the GermEval 2014 dataset, where the tags constitutes four major types, i.e., LOC, PER, ORG and OTH, and their sub-types which end with "-deriv" or "-part" using a BIO tagging scheme. The results in

[2] The hyperparameters for scheduling PL were chosen via cross validation.

Table 1: Effect of word embeddings

Initialization	P	R	F_1
Random	69.67	54.19	60.96
Pretrain	68.39	69.27	68.82

Table 2: Effect of Pseudo Label as a regularizer

Learning scheme	P	R	F_1
Sup. learning ($\beta_t = 0$)	68.39	69.27	68.82
Semi-sup. ($\beta_{\max} = 2$)	77.08	68.40	72.48

Table 1 and 2 are reported in terms of the official metric, namely M1 (See (Benikova et al., 2014a)), in the GermEval 2014 Shared Task.

4.1 Details of Training

We evaluated the proposed method with the following hyperparameter settings over the number of hidden units $F = 300$, the dimension of capitalization features $d_{cap} = 3$, the dimension of named entity features $d_{ne} = 10$, window size $k_w = 5$, $\alpha = 0.5$, a fixed learning rate 0.01 for SGD with AdaGrad (Duchi et al., 2011). In addition, we used length normalization over all embeddings such that $\|x\| = 10$ to prevent overfitting. For the transfer function in Eq.1, *ReLU*, $f(x) = \max(0, x)$, is used. The feature matrices $\mathbf{L}^{(caps)}$ and $\mathbf{L}^{(ne)}$ were initialized randomly.

4.2 Importance of Word Embeddings

We used *word2vec* (Mikolov et al., 2013) for learning word embeddings because of its efficiency.[3] We set the dimension of word embeddings d_w to 128 and the size of vocabulary $|V|$ is about 4M which yields the feature matrix $\mathbf{L}^{(w)} \in \mathbb{R}^{128\times 4M}$. We run the *word2vec* for 10 epochs with a fixed learning rate 0.01 on approximately 87M sentences from a German Wikipedia dump, LCC, and SDeWac (Faaß and Eckart, 2013).

The results of using pretrained word embeddings on unlabelled data in comparison to random initialization are shown in Table 1. We observed that NNs using pretrained word embeddings perform much better in terms of *recall*.

4.3 Effect of Semi-Supervised Learning

We evaluated our proposed approach for nested NER. The results of this experiment are shown in Table 2. The semi-supervised approach outperforms the purely supervised one. We observe that learning with pseudo labels reduces the number of false positives which results in higher precision. In particular, the number of predictions in the top-level resulting from the supervised approach is 2738 while the semi-supervised approach yields 2378 predictions. Interestingly, we also observe performance improvement on LOC and ORG as well as the smaller classes including OTH and "deriv"- and "part"-classes, but not all of them.

Table 3: The System Performance on Unseen Data

Metrics	P	R	F_1
M1	76.76	66.16	71.06
M2	78.09	67.31	72.30
M3 (1^{st} level)	77.93	68.52	72.92
M3 (2^{nd} level)	57.86	37.86	45.77

4.4 Results of GermEval 2014 Shared Task

The proposed method was submitted to the GermEval 2014 Named Entity Recognition Shared Task. Our system called **PLsNER** was ranked at 5^{th} and the scores are shown in Table 3. More results and comparisons with other systems can be found in (Benikova et al., 2014a).

5 Conclusions

We proposed a neural network architecture, which is capable of learning from top-level NEs and nested NEs jointly in nested NER. By making use of unlabelled data in a semi-supervised fashion, we also demonstrated its effectiveness when a small number of training examples are provided.

Our experiments show that the use of word embeddings improves *recall* compared to random initialization. Pseudo labels make it possible to get more *precise* predictions. Additionally, our system performs pretty well on unseen data without use of language-dependent feature engineering steps.

Acknowledgments

This work has been supported by the German Institute for Educational Research (DIPF) under the Knowledge Discovery in Scientific Literature (KDSL) program.

[3] https://code.google.com/p/word2vec/

References

Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Pado. 2014a. GermEval 2014 Named Entity Recognition: Companion Paper. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014b. NoSta-D Named Entity Annotation for German: Guidelines and Dataset. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa. 2011. Natural Language Processing (almost) from Scratch. *Journal of Machine Learning Research*, 12:2493–2537.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12:2121–2159.

Gertrud Faaß and Kerstin Eckart. 2013. SdeWaC–a Corpus of Parsable Sentences from the Web. In *Language processing and knowledge in the Web*, pages 61–68. Springer.

Yves Grandvalet and Yoshua Bengio. 2005. Semi-supervised Learning by Entropy Minimization. In *Advances in Neural Information Processing Systems 17*, pages 529–536.

Dong-Hyun Lee. 2013. Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks. In *Workshop on Challenges in Representation Learning, ICML*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Uwe Quasthoff, Matthias Richter, and Chris Biemann. 2006. Corpus Portal for Search in Monolingual Corpora. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, pages 1799–1802, Genoa.

Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. 2005. Semi-Supervised Self-Training of Object Detection Models. In *Proceedings of the Seventh IEEE Workshops on Application of Computer Vision*, WACV-MOTION '05, pages 29–36.

Adapting Data Mining for German Named Entity Recognition

Damien Nouvel
Université François Rabelais Tours
Laboratoire d'informatique
Tours, France
damien.nouvel@limsi.fr

Jean-Yves Antoine
Université François Rabelais Tours
Laboratoire d'informatique
Tours, France
jean-yves.antoine@univ-tours.fr

Abstract

In the latest decades, machine learning approaches have been intensively experimented for natural language processing. Most of the time, systems rely on using statistics within the system, by analyzing texts at the token level and, for labelling tasks, categorizing each among possible classes. One may notice that previous symbolic approaches (e.g. transducers) where designed to delimit pieces of text. Our research team developped mXS, a system that aims at combining both approaches. It locates boundaries of entities by using sequential pattern mining and machine learning. This system, intially developped for French, has been adapted to German.

1 Introduction

In the 90's and until now, several symbolic systems have been designed that make intensive use of regular expressions formalism to describe Named Entities (NEs). Those systems combine external and internal evidences (McDonald, 1996), as patterns describing contextual clues and lists of names per NE category. Those systems achieve high accuracy for NE Recognition (NER), but, because they depend on the handcrafted definition of lexical ressources and detection rules, their coverage remains an issue.

To address NER, machine learning usually states the problem as categorizing words that belong to a NE, taking into account various clues (features) in a model that is automatically parametrized by leveraging statistics from a training corpus. Among these methods, some only focus on the current word under examination (maximum entropy, SVM) (Borthwick et al., 1998), while others also evaluate stochastic dependencies (HMM, CRF) (McCallum and Li, 2003; Ratinov and Roth, 2009). Most of the time, those approaches output the most probable sequence of labels for a given sentence. This is generally known as the "labeling problem", applied to NER.

Many approaches rely on pre-processing steps that provide additional information about data, often Part-Of-speech (POS) tagging and proper names lists, to determine how to automatically tag texts (Ratinov and Roth, 2009). Recently, data mining techniques (Freitag and Kushmerick, 2000) have been experimented, but we are not aware of work that goes beyond the step of extracting patterns for NER.

Our system, mXS[1] (Nouvel et al., 2014), automatically mines patterns and use them as features for machine learning. It focuses on boundaries of NEs, as beginning or ending tags to be inserted. Internally, the system considers each tag delimiting a NE as an item of interest and extracts detection rules (which may be used as feature but also may be read by humans). To the best of our knowledge, this way of combining symbolic and machine learning approaches is original in the framework of NER. It obtained satisfying results

[1] https://github.com/eldams/mXS

during the ANR ETAPE of the ANR French research agency evaluation campaign, ranked 3rd or 2nd among 10 particpants. This paper presents our adaptation of mXS to German.

2 Coding, Preprocessings and Lexicon

2.1 Coding NEs beyond BIO Format

As previously mentioned, most of the approaches for doing NER rely on labelling tokens of a text. This leads to representations as illustrated in Figure 1 where each token is assigned a dedicated class. Machine learning approaches are known to be efficient to solve this kind of problem. Our main concern about this representation is that it is now mandatory to classify all tokens within a named entity, even underspecific tokens such as `für/I-ORG`.

As a result, mXS uses internally a different coding to represent NE tokens: only beginning and ending of NEs are explicitly mentionned, in a XML-like fashion, e.g. `<PER> Cartier </PER>`. Our goal is then to discover the correct positions where NE tags have to be inserted, as showed in Figure 2. This approach doesn't prevent to use machine learning techniques, avoids the artificial split of NE classes (e.g. `B-XXX` and `I-XXX`) and can be used in combination with sequential data mining techniques.

2.2 Morphosyntax

Initial preprocessings and linguistic analysis are done using TreeTagger (Schmid, 1994), that conjointly tokenizes, lemmatizes and assigns POS to each token. Our first experiments demonstrate that this software gives sufficient clues, especially by identifying proper names, to ground our system. We use this information, as gradual generalizations for building representation of texts. Consider for instance this sentence from the GermEval training corpus:

```
Der <LOC> Queen <PER> Sirikit </PER>
Park </LOC> ist ein Botanischer Garten
```

Here, Botanischer is progressively generalized as `botanisch` (lemma) then `ADJA` (adjective POS). This incremental generalization is described by `ADJA/botanisch/Botanischer` where the `/` symbol is used as a specialization operator. Our text mining process is able to consider for any token all possible generalizations over this hierarchy[2]. The sentence is now represented as:

```
ART/die/Der <LOC> NN/Queen/Queen
<PER> NN/Sirikit/Sirikit </PER>
NN/Park/Park </LOC> VAFIN/sein/ist
ART/eine/ein ADJA/botanisch/Botanischer
NN/Garten/Garten
```

As data mining process is aimed at extracting generic patterns, we exclude surface variations (but keep their lemmas) and lexicalization of proper names (to avoid overfitting) when preprocessing training corpus:

```
ART/die <LOC> NN/Queen <PER> NN/Sirikit
</PER> NN/Park </LOC> VAFIN/sein
ART/eine ADJA/botanisch NN/Garten
```

The French version of mXS includes many dedicated adaptations to improve recognition of specific linguistic expressions. The German version of mXS that participates to GermEval does not include such useful improvements.

2.3 Lexicon

In the experiments presented in Section 4, the baseline system does not use any lexicon, and thus only relies on morphosyntax analysis. To improve performance, we also considered three proper noun lexicons as additional resources (Table 1): ST is extracted from FreeBase ; IP and IW are gross-grained and fine-grained versions of a lexicon extracted from Wikipedia (Savary et al., 2013). They implement usual classes for NER as antrhroponyms, toponyms, first names, last names, organizations, etc.

Lexicon	Categories	Entries
ST	5	497 093
IP	7	33 167
IW	118	33 167

Table 1: System lexicons number of classes and entries

Those lexicons provide another possible level of generalization. As it is more related to semantic properties of tokens, this information will be considered as the top level to generalize tokens. mXS also supports multiword expressions and ambiguity at any level: semantic categories

[2]Besides, as it is not a column format, the number of possible generalizations may vary from one token to another

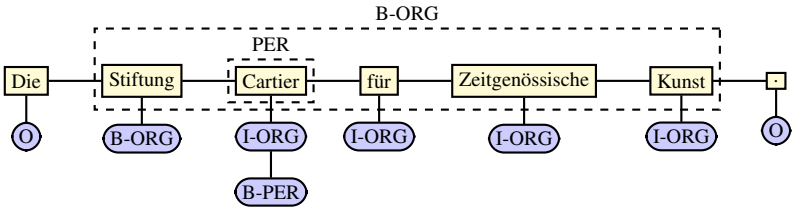


Figure 1: Annotation as a labelling task

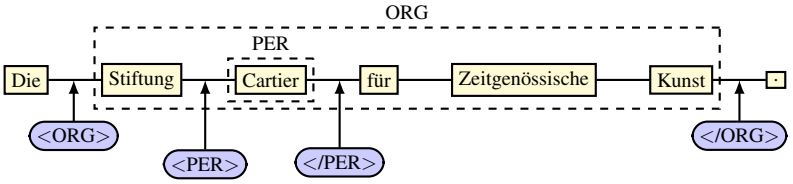


Figure 2: Annotation as an annotation task

provided by lexicons may be assigned to multiple tokens, and each token may receive multiple categories. Using those lexicons adds information:

```
  -/ART/die/Der <LOC>
Organizations/NN/Queen/Queen
<PER> -/NN/Sirikit/Sirikit
</PER> -/NN/Park/Park </LOC>
-/VAFIN/sein/ist -/ART/eine/ein
Locations/ADJA/botanisch/Botanischer
Locations/NN/Garten/Garten
```

Furthermore, for TreeTagger categories `NN` and `NE`, suffixes with a size of 3 or 4 characters are also considered as an intermediate generalization level, e.g. `Locations/NN/Garten` now becomes `Locations/NN/SUFF:ten/SUFF:rten/Garten`. This also illustrates how hierarchical sequential mining can easily fit special needs (e.g. language or task adaptation of preprocessings).

3 Sequential Data Mining to extract Patterns as Features

Mining techniques are applied on the information provided by preprocessings. The data miner within mXS proceeds in a supervised level-wise fashion to extract generalized sequential patterns (Agrawal and Srikant, 1995) that are correlated to NE tags. To limit complexity, the search is limited by criterions such as minimum support (frequency), minimum confidence (regarding the presence of NE tags) and redundancy within patterns. Extracted patterns are supposed to be valuable clues for detecting NE boundaries. Due to a lack of space, the mining process will not be detailed in this paper, further information can be found in (Nouvel et al., 2014).

mXS implements hierarchical mining: patterns are sequences of diversely generalized natural language tokens and enriched data and NE tags. Here are some examples of extracted patterns:

```
<PER> NE ART NN/SUFF:ung
<LOC> CITY/NN APPR/in REGION/NE </LOC>
<PER> NE NN APPR CITY </LOC>
```

The extracted patterns are used as features by a maxent classifier, provided by the scikit-learn toolkit (Pedregosa et al., 2011) that estimates, at any position of a sentence, the probability to insert tags given the patterns. using a Viterbi algorithm, the decoding step combines individual probabilities to select annotation that maximizes likelihood. The advantage of this approach, besides avoiding the artificial split of B- and I- of BIO format, is that it can insert multiple tags at a given position, enabling recursive annotation as required by the GermEval campaign.

4 Experiments and Results

We assess the usefulness of the extracted patterns for NER, by selecting them at different thresholds of support and confidence. Table 2 shows that best score are obtained with low support (5) and medium confidence (10%). Around 17000

patterns are extracted with these parameters. The comparison with situations where pattern features are not used ("inf") shows that patterns always lead to better performances, reaching a maximum increase of +2.5% of the overall f-score.

supp	conf%	rules	fscore%	prec%	rec%
5	5	21 620	59.50	76.44	48.71
5	10	17 268	**59.91**	76.76	**49.13**
5	50	7 512	58.87	76.87	47.70
10	5	9 505	59.62	76.82	48.71
10	10	7 460	59.55	76.68	48.67
10	50	3 108	58.53	76.80	47.28
50	5	1 283	59.41	77.37	48.22
50	10	972	59.35	**77.42**	48.11
50	50	359	58.35	77.03	46.96
inf	inf	0	57.41	76.01	46.12

Table 2: Score without lexicon

We investigated the benefits of using three lexicons, separately or jointly. As displayed in Table 3, using them always lead to significant improvement. Unfortunately, combining them degrades performances (we assume that those resources are not as complementary as expected).

lex	supp	conf%	fscore%	prec%	rec%
none	5	10	59.91	76.76	49.13
ST	50	50	**62.97**	**80.63**	**51.66**
IP	10	10	61.07	78.83	49.84
IW	5	20	60.38	78.10	49.22
All	50	10	62.71	80.61	51.31

Table 3: Score depending on lexicon

We built our final system using only the ST lexicon, which provided the best score (63.16), each run being a combination of frequency and confidence parameters. Official results in Table 4 are close to what has been obtained on the development dataset and unfortunately confirmed our very high precision but unsufficient recall: our system is ranked 7th out of 11. We suspect overfitting and conducted additional experiments for fine-tuning maxent regularization parameter. For the moment, this leads to a better f-score (64.19) over the official test data, without clarifying the question of the strong difference between precision (80.76) and recall (53.26).

supp	conf%	fscore%	prec%	rec%
5	10	61.63	79.05	50.5
10	50	62.29	80.46	50.81
50	50	**62.39**	**80.62**	**50.89**

Table 4: Final scores

5 Conclusion

This paper shows how to use data mining in an original way (separate detection of NE boundaries instead of BIO tagging) to implement a rather efficient multilevel named entity recognition system. Adapting mXS from French to German was quite easy, thanks to the availability of resources. Obviously, this version of mXS lacks linguistic adaptations specific to German, what prevent us to reach an optimal level of performance. Nevertheless, we reached our main goal, which was to assess the reliability of our original approach on another language using similar preprocessings steps and our generic pattern mining implementation.

Acknowledgments

Thanks to people from LIMSI-CNRS and ANR ETAPE for endorsing our work on NER.

References

Rakesh Agrawal and Ramakrishnan Srikant. 1995. Mining sequential patterns. In *ICDE*, pages 3–14.

Andrew Borthwick, John Sterling, et al. 1998. Exploiting diverse knowledge sources via maximum entropy in NER. In *WVLC'1998*.

Dianne Freitag and Nicholas Kushmerick. 2000. Boosted wrapper induction. In *WMLIE*.

Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *CONLL'2003*.

David D. McDonald. 1996. Internal and external evidence in the identification and semantic categorization of proper names. *CPLA*, pages 21–39.

Damien Nouvel, Jean-Yves Antoine, and Nathalie Friburger. 2014. Pattern mining for named entity recognition. *LNCS/LNAI Series*, 8387i.

Fabian Pedregosa, Gaël Varoquaux, et al. 2011. Scikit-learn: Machine learning in Python. *J. of Machine Learning Research*, 12:2825–2830.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in NER. In *CONLL'2009*, pages 147–155, Stroudsburg, PA, USA. ACL.

Agata Savary, Leszek Manicki, and Malgorzata Baron. 2013. Populating a multilingual ontology of proper names from open sources. *J. Language Modelling*, 1(2):189–225.

Helmut Schmid. 1994. Probabilistic POS tagging using decision trees. In *NMLP*.

Named Entity Recognition for German Using Conditional Random Fields and Linguistic Resources

Patrick Watrin
EarlyTracks SA
Louvain-la-Neuve
Belgium

Louis de Viron
EarlyTracks SA
Louvain-la-Neuve
Belgium

Denis Lebailly
EarlyTracks SA
Louvain-la-Neuve
Belgium

Matthieu Constant
Université Paris-Est
Marne-la-Vallée
France

Stéphanie Weiser
EarlyTracks SA
Louvain-la-Neuve
Belgium

Abstract

This paper presents a Named Entity Recognition system for German based on Conditional Random Fields. The model also includes language-independant features and features computed form large coverage lexical resources. Along side the results themselves, we show that by adding linguistic resources to a probabilistic model, the results improve significantly.[1]

1 Introduction

These last few years, models based on Conditional Random Fields (CRF) have shown interesting achievements for Named Entity Recognition (NER) tasks. However, most of the experiences carried out also show a lack of lexical coverage. To counterbalance this lack, two main kinds of strategies have been designed: the use of gazetteers and of clustering techniques. Both lead to a significant improvement of the results. For a review of these techniques, see (Tkachenko and Simanovsky, 2012). In the work presented here, we have opted for a more linguistic approach, close to the gazetteers: we included lexical resources as new features for a model based on CRF and measured their impact. This kind of approach has already been proven successful for a Part-of-Speech tagger by Constant and Sigogne (2011).

This work took place in the framework of the GermEval Named Entity Recognition Shared Task[2] and is therefore applied to German. However, this approach has already been implemented for English, French and Dutch.

The characteristics of the GermEval tagset are presented in section 2. In section 3 is described our system for named entity recognition based on CRF and the adaptations we suggest for this kind of model. Section 4 presents the linguistic resources we added. Finally, our experiments and the results we obtained are presented in section 5.

2 GermEval Characteristics

2.1 Tagset

The tagset defined for the GermEval shared task (Benikova et al., 2014b) consists of four main classes. The class *Person* (1) includes person names but also nicknames and fictional characters names. The class *Organisation* (2) contains all kind of organisations, companies, and also festivals, music bands, etc. The *Location* class (3) is made for all kind of places: cities, countries, planets, churches, etc. The class *Other* (4), is the widest one as it includes a large variety of items: movies and books titles, languages, websites, market indexes etc.

These four main classes have two subclasses each: *deriv* and *part* (LOCderiv, OTHderiv, PERderiv, ORGderiv, LOCpart, OTHpart, PERpart, ORGpart). The *deriv* one is used to tag items that are derived from named entities. Most of the times they are adjectives such as *asiatischen* (*asian*). The *part* one is made for named entities that are included in a larger token, in compound

[2]https://sites.google.com/site/germeval2014ner/home

forms. As the German language is agglutinative, this happens quite often, without diacritical marks (*Bundesligaspiele*) .

2.2 Entities Embedding

Another specificity of the GermEval task is that nested entities are allowed. For example, the film title *Shakespeare in Love* must be tagged OTH but it must also contain an inner tag PER for *Shakespeare*. The tagger we developed therefore needed to be adapted to include this particularity.

3 Conditional Random Fields

As presented in (Lafferty, 2001), CRF define a framework for building probabilistic models that are able to split and tag sequences of data. Since they exist, CRF have lead to many works in Natural Language Processing (*e.g.* Constant and Sigogne (2011)) and more specifically in NER (*e.g.* Finkel et al. (2005) and Klein et al. (2003)).

3.1 Standard Approach

In practice, the probability of a sequence of labels depends on a set of features that are representative of the observation sequence (*i.e.* the tokens). Most of these features are language-independent and limited to local observations. CRF systems generally use a set of generic features, that we present in table 1.

These features are language-independent. However, some characteristics of the language can be in conflict with one or more features. For example, the feature that represents the presence or absence of a capital letter is less pertinent for German – where many words begin with a capital letter – than for other languages.

3.2 Hybrid approach

The statistical models are limited to their training corpus and therefore their lexical coverage is often not large enough. Many works have tried to compensate for this weak coverage to help the classification of unseen words. Faruqui and Padó (2010) and Finkel et al. (2005) suggest to add a distributional similarity feature trained on a very large corpus. The hypothesis of a strong correlation between the terms of a same distributional class is the basis of this feature. Faruqui and Padó (2010) show very interesting results for German,

Feature	Explanation
$\dots w_{-1} w_0 w_1 \dots$	tokens
lowercase	token in lowercase
shape	token in a Xx form
isCapitalized	is the token capitalized?
prefix(n)	n first letters of the token (1 to 4)
suffix(n)	n last letters of the token (1 to 4)
hasHyphen	does the token contain hyphens?
hasDigit	does the token contain digits?
allUppercase	is the token uppercase only?

Table 1: Language-independent features

Feature	Explanation
pos	Token PoS-tag
containsFeature(x)	Does the token belong to the semantic class x?
sac	Semantic ambiguity class *i.e.* all possible classes for the token

Table 2: Lexical features

with an increase of 6-7% for precision and 12-13% for recall.

In parallel to this method, other studies suggest the use of external lexical resources (Nadeau and Sekine, 2007; Kazama and Torisawa, 2007; Constant and Sigogne, 2011). Indeed, a simple way to decide if a sequence of tokens corresponds to a named entity is to check in a dictionary. Today, many multilingual encyclopedic resources are available online and facilitate the construction of these dictionaries (DBPedia, Yago, FreeBase...). To integrate the information of these dictionaries in our model, we have defined 3 types of features, that are presented in table 2, where the classes correspond to the different classes of the GermEval tagset. The linguistic resources we used and their impact are presented in section 4 and 5.

4 Adding Linguistic Resources to the Model

The linguistic resources we used are divided into two types: dictionaries (word lists including morphological data) and grammars made of transducers created with the software Unitex[3]. The objective of these resources is to counterbalance the lack of lexical coverage due to the training corpus.

4.1 Dictionaries

We use two kinds of dictionaries. First, we use a general language dictionary of German, that we adapted from the resources created by Daniel Naber[4], using Morphy[5]. It contains lemmas, in-

[3]http://www-igm.univ-mlv.fr/ unitex/

[4]http://danielnaber.de/morphologie/

[5]http://www.wolfganglezius.de/doku.php?id=cl:morphy

Dictionary	Nb. entries
Morphy	749.212
Persons	1.266.390
Places	200.392
Places *deriv*	2.642
Organisations	648.273
Others	2.617.902

Table 3: Number of entries by dictionary

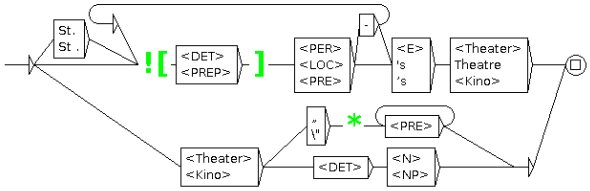


Figure 1: Transducer for matching Theatres such as *Berlin's Theater*

flectional variations and part-of-speech tags. The second type of dictionaries are useful for data that can be fully listed, such as countries for example. We created dictionaries for most of the entities that needed to be extracted using free resources such as Freebase[6]. We also created dictionaries for the *deriv* entities to follow the GermEval guidelines. Table 3 gives the number of entries for each dictionary.

4.2 Local Grammars

Local grammars that we created using Unitex transducers (Paumier, 2003) are efficient for entities that can vary more or are difficult to fully list. For example, a grammar can be defined to describe all kind of universities or theatres names, as it is shown in the figure 1.

These grammars can also handle German specificities such as concatenation of words. Some specific transducers have been made to cover the *part* entities (when an entity is included in a larger token as *Hamiltonoperator* for example). Our grammar library contains 9 main graphs (one for each category, one for each *deriv* category and one for all *part* entities) and around 20 subgraphs.

5 Experiments and Results

In this section, we present our experiments and put our results in balance with those of the other participants to the GermEval task. The table 4 shows the results obtained by all the systems that have participated to the GermEval 2014 Shared Task. We rank number 4, out of 11 models competing. The table 5 presents the results we obtained with two models: the simple CRF model and the model enriched by the lexical resources. The four metrics we use are explained by Benikova et al. (2014a).

[6]http://www.freebase.com/

Model	Precision	Recall	F_1
ExB	78.07	**74.75**	**76.38**
UKP	79.54	71.1	75.09
MoSTNER	79.20	65.31	71.59
EarlyTracks	79.92	64.65	71.48
PLsNER	76.76	66.16	71.06
DRIM	76.71	63.25	69.33
mXS	**80.62**	50.89	62.39
Nessy	63.57	54.65	58.78
NERU	62.57	48.35	54.55
HATNER	65.62	43.21	52.11
BECREATIVE	40.14	34.71	37.23
Median	76.71	63.25	69.33

Table 4: Results obtained by all the participants to the GermEval 2014 NER Shared Task (Strict Metric)

Model	Metric	Precision	Recall	F_1
CRF	M-Strict	77.14	61.56	68.47
	M-Loose	77.89	62.15	69.14
	M-Outer	77.57	63.89	70.07
	M-Inner	68.38	33.59	45.05
CRF+LING	M-Strict	79.92	64.65	71.48
	M-Loose	80.55	65.16	72.04
	M-Outer	80.44	66.98	73.10
	M-Inner	70.00	36.70	48.15

Table 5: Impact of adding linguistic resources to a CRF model

Our results are interesting because they show that by adding lexical resources and grammars as new features to our model, the results are improved by 3.01% for the strict metric, which is significant. This number should keep rising while the resources increase.

Table 6 shows the results obtained for each outer class and each inner class and the improvement made with lexical resources. As the class OTH is very versatile, it obtains less good results than the other classes. Furthermore the entity classes *part* and *deriv*, as well as the inner-classes, are less represented in the training set and therefore also reach less good results. The classes ORG, LOC and PER which can rely on external lexical resources obtain better results.

Entity	M-Outer			M-Inner		
	Occ.	**CRF**	**CRF+**	**Occ.**	**CRF**	**CRF+**
PER	1639	76.63	80.20	82	4.49	10.87
ORG	1150	63.54	66.34	41	8.51	8.89
LOC	1706	75.54	79.36	210	56.09	56.99
OTH	697	50.51	52.46	7	0.00	0.00
PERpart	44	16.00	12.24	4	40.00	40.00
ORGpart	172	56.39	58.61	1	0.00	0.00
LOCpart	109	55.49	54.97	5	0.00	0.00
OTHpart	42	16.33	25.00	1	0	0
PERderiv	11	16.67	0.00	4	0.00	0.00
ORGderiv	8	22.22	22.22	1	0	0
LOCderiv	561	78.31	80.15	159	54.12	59.46
OTHderiv	39	47.46	47.62	0	0	0
Global	**6178**	**70.07**	**73.10**	**515**	**45.05**	**48.15**

Table 6: For each outer and inner entity: number of occurrences in the evaluation corpus and F_1 for the simple CRF and the enriched CRF

6 Conclusion

In this paper, we presented our Named Entity Recognizer for German. We achieve a global F-measure of 71.48% on the GermEval evaluation corpus with the complete tagset. In parallel, we evaluated the impact of using linguistic resources as an input to the statistical model: it improves the results by 3.01% for the strict metric. As a next step, to increase this impact, the dictionaries, that are still in an early stage, should be enhanced: they have been automatically gathered and could use a manual correction to avoid erroneous entries. In addition, we will try to find other precise dictionaries and enlarge the grammars to improve the recall, in particular to cover more completely the *Others* class.

Another possible way of improving our system would be to combine our linguistic approach to a clustering strategy.

References

Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Pado. 2014a. Germeval 2014 named entity recognition: Companion paper. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014b. Nosta-d named entity annotation for german: Guidelines and dataset. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Matthieu Constant and Anthony Sigogne. 2011. Mwu-aware part-of-speech tagging with a crf model and lexical resources. In *Proceedings of the Workshop on Multiword Expressions: From Parsing and Generation to the Real World*, MWE '11, pages 49–56, Stroudsburg, PA, USA. Association for Computational Linguistics.

Manaal Faruqui and Sebastian Padó. 2010. Training and evaluating a german named entity recognizer with semantic generalization. In *Proceedings of KONVENS 2010*, page 129. Semantic Approaches in Natural Language Processing.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.

Jun'ichi Kazama and Kentaro Torisawa. 2007. Exploiting wikipedia as external knowledge for named entity recognition. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 698–707. Citeseer.

Dan Klein, Joseph Smarr, Huy Nguyen, and Christopher D. Manning. 2003. Named entity recognition with character-level models. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 180–183. Edmonton, Canada.

John Lafferty. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001)*, pages 282–289. Morgan Kaufmann.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Sébastien Paumier. 2003. *De la reconnaissance de formes linguistiques à l'analyse syntaxique*. Ph.D. thesis, Université de Marne-la-Vallée, July.

Maksim Tkachenko and Andrey Simanovsky. 2012. Named entity recognition: Exploring features. In Jeremy Jancsary, editor, *Proceedings of KONVENS 2012*, pages 118–127. ÖGAI, September. Main track: oral presentations.

NERU: Named Entity Recognition for German

Daniel Weber, Josef Pötzl
CIS, Ludwig Maximilian University, Munich
forwarding@go4more.de
j.poetzl@campus.lmu.de

Abstract

In this paper [1], we present our Named Entity Recognition (NER) system for German – NERU (Named Entity Rules), which heavily relies on handcrafted rules as well as information gained from a cascade of existing external NER tools. The system combines large gazetteer lists, information obtained by comparison of different automatic translations and POS taggers. With NERU, we were able to achieve a score of 73.26% on the development set provided by the GermEval 2014 Named Entity Recognition Shared Task for German.

1 Introduction

Generally, named entities (NEs) are phrases that represent persons, organizations, locations, dates, etc. For example, the German sentence "*Frau Maier hat einen Toyota aus Amerika gekauft.*" contains three named entities *Frau Maier*, which refers to a person, *Toyota*, referring to an organization and *Amerika*, marking a location. Embedded NEs may also be present, for example: *Troia - Traum und Wirklichkeit* is a NE, which contains an embedded NE of type location – *Troia*.

In this paper, we describe NERU, which is a rule-based system for NER for German that was developed in the context of the GermEval 2014 NER Shared Task that specifically targets only this language. Thus, NERU aims to identify not only flat NE structures, but as well embedded ones. As described by Benikova et al. (2014b), the maximal level of embedding for the GermEval 2014 task is one named entity. The main targeted types are PER (person), LOC (location), ORG (organization) and OTH (other) with two possible subtypes relevant for all four groups – deriv and part. The latter leads to a combination of 12 different NE types.

Following, in section 2, we discuss the motivation behind GermEval 2014 and the state-of-the-art approaches to NER focusing on the language important for this task – German. Then, in section 3, we provide more details on the structure of NERU and the approach we used. In section 4, we present the performance of the system on the development data provided by the GermEval 2014 shared task. Finally, in section 5, we conclude our work.

2 Related Work

NER is an important subtask of a wide range of Natural Language Processing (NLP) tasks from information extraction to machine translation and often even requires special treatment within them (Nagy T. et al., 2011). GermEval's goal is, however, to consider NER proper and to advance the state-of-the-art of this task for a particular language – German. This language has been rarely the focus within previous NER research, which mostly explores English. The CoNLL-2003 Shared Task on Language-Independent NER (Tjong Kim Sang and De Meulder, 2003) addressed this problem and included German as one of its targets, although, in general, multilin-

guality was the objective.

While the majority of NER so far was concentrating almost only on flat NE structures (Tjong Kim Sang and De Meulder, 2003; Finkel and Manning, 2009), one of the main goals of GermEval is also to push the field of NER towards nested representations of NEs. Independent of the NE representation itself, there are many different approaches to tackle this task, for example, by using machine-learning techniques, such as Hidden-Markov-Models (Morwal et al., 2012), rule-based (Riaz, 2010) or even a combination of both (Nikoulina et al., 2012). NE recognition utilizing a hybrid approach has also been performed by Saha et al. (2008), who presented a set of handcrafted rules together with the use of gazetteer lists which were transliterated from English to Hindi with their own transliteration module.

As German significantly differs from other languages regarding capitalization or syntax in general, some of the common approaches, specifically on English, can not be transferred to German automatically. Thus, in the context of GermEval, we concentrate mostly on handcrafted rules as well as information from external NER tools. The full pipeline of the NERU system is presented in more detail further in section 3.

3 The NERU System

NERU's pipeline is structured as follows: In a first step, we use vast gazetteer lists to attain first suggestions for NEs (see section 3.1). Secondly, we utilize automatic translation tools to find matches occurring in various languages (described in section 3.2). Thirdly, the results of the TreeTagger (see section 3.3), the Stanford NE Recognizer (see section 3.4) and examining contexts of NE's (see section 3.5) are then taken into consideration. The combination results to a cascade of different methods that provide a set of suggestions for the NEs in the data. In a last step, we revise this set and modify it by removing and altering its entries with a number of manually crafted rules (see section 3.6).

3.1 Gazetteers

Gazetteers are predefined word lists which represent standard sources for NER as they contain NEs, such as names, organizations and locations marked for their correct category. So far, gazetteers were widely employed for tackling this task (Kazama and Torisawa, 2008; Jahangir et al., 2012; Alotaibi and Lee, 2013). NERU also employs gazetteers (mainly lists of locations and persons), which were collected from the German Wikipedia[2] and then manually extended.

One of the biggest problems in NER is resolving ambiguity. If all NEs are unambiguously identifiable, a large gazetteer would be sufficient. In natural language, however, there are context-sensitive terms, such as *California Institute of Technology*, which can on the one hand appear as a location and on the other as an organization. The decision as to which category the Named Entity shall be assigned depends solely on its textual environment.

3.2 Preclusion Through Translation

To deal with false-positives generated with the use of gazetteers, more sophisticated methods are needed to perform viable NER. In order to also consider the textual environment of the tokens, we make use of machine translation (MT). In fact, translations of NEs often leads to the use of the same surface form in both languages, specifically most proper names are not affected by the translation procedure. Therefore, we assume that all tokens that do not change within translation are reasonable NE candidates.

The Google Translate API[3] is used for translating the German data into English. For stopwords that are present in both languages, which should not be marked as NEs, we incorporated a list created by the intersection of the lists of stopwords from both English and German.

3.3 TreeTagger

To provide further suggestions for NEs, we employ the TreeTagger (Schmid, 1994; Schmid, 1999), which is a robust POS tagger for German reaching state-of-the-art performance. The tagger may also be partially used as a recognizer when the POS tags for proper names (*NE*) are employed. Hence, all tokens tagged with the *NE* tag are also considered as NE candidates.

[2] https://de.wikipedia.org

[3] https://developers.google.com/translate

3.4 Stanford NER

In the search for a wider source of diverse suggestions for the NEs in the data, we embedded the Stanford NER[4] in our system to find additional candidates for NEs. It is very robust in detecting NEs, however being restricted to only one type of NE – PER. All tokens marked as NE by the Stanford NER are again used as NE candidates by NERU.

3.5 Context Frequency and Probability

Using the GermEval training data, we also detect potential NEs by observing their type and frequency of contexts. If token t is marked by a NE tag (e.g. B-LOC, I-PER, etc.), we extract a NE-trigram (t_{-1}, t, t_{+1}) for it. Frequency counts of the trigrams are then collected and the ones occurring less than 5 times are ignored. Following, the probability of a token in a specific context is calculated. Only tokens that have a probability > 0.5 of being in that context are marked as NEs.

Assuming a token sequence "*der philippinischen Hauptstadt*" is encountered, "*philippinischen*" would be tagged as B-LOCderiv. If there are different options for a NE tag in this context, the option with the highest probability is chosen.

3.6 Rule-Based Filtering

In sections 3.1 through 3.4, we presented a number of different approaches, which we used for the identification of NEs in the data. This cascade of modules, however, results to a generously tagged dataset including suggestions for as many NEs as possible. In order to reduce this set, in the last step of NERU's pipeline, we process the output with the help of a collection of handcrafted rules. An additional set of rules is also used that relies only on the information provided by the gazetteers and manually created lists of abbreviations.

3.7 Rules for Person NEs

To identify NE of the type PER, we examine contexts and tokens we categorized as trigger words, such as nobiliary particles, honorary or heredity titles, etc. For example, Roman numerals may indicate a person (e.g. *Karl IV*), similar to the generational title "*Jr.*", which may also appear following the candidate NE. Additionally, when particles, such as "*von*" or "*de*" are found between two or more NEs of the type PER or the special case that a NE of the type LOC is perceived right after "*von*" (*of*), the latter are combined to one single span, for example "*Wilhelm Friedrich Ludwig von Preußen*".

[4] http://nlp.stanford.edu/software/CRF-NER.shtml

3.8 Rules for Organization NEs

For the identification of organizations, we looked for special characters like "*&*" between NEs of type PER (e.g. *Kanzlei Heigl & Hartmann*). We furthermore deduce organization names from common abbreviations. If a token is found, which is marked as a LOC or a PER and its preceding token is a common abbreviation (e.g. *AC*, *TSV* etc., which we check against a manually created list of common abbreviations), then the whole sequence indicates a NE of type ORG (e.g. *FC Barcelona*).

In a similar way, the abbreviations for a type of organization, such as "*GmbH*", "*Comp.*", "*KG*" are also used as indicators for NEs of type ORG. Such tokens or their attributed NEs are combined with any closely preceding NE of type ORG or PER. It is not distinguished between the types ORG and PER, as we consider organization names like "*Wortmann AG*". We investigate the preceding tokens until a token which has been tagged as ORG or PER is found, unless the examined sequence is larger than 5 tokens. In this case, the 5th token is chosen automatically. For example, if "*Bandidos Kapital und Invest AG*", is considered and only the token "*Bandidos*" is already tagged as a NE of type ORG, the identification of the abbreviation "*AG*" would impose the marking of the full span as NE of type ORG.

3.9 Rules for Location NEs

In order to recognize location names, we look for specific character patterns, such as "*straße*" (street) in the tokens (e.g. *Leopoldstraße*). Once more, we investigated the contexts to properly find connected sequences. For example, when a number is preceded by a NE of type LOC, the number is also included into the NE sequence (e.g. "*Dachauer Straße 24*").

setting	strict				loose				outer				inner			
	Acc.	**P**	**R**	**F1**	**Acc.**	**P**	**R**	**F1**	**Acc.**	**P**	**R**	**F1**	**Acc.**	**P**	**R**	**F1**
CF	93.58	15.32	10.67	12.58	93.59	15.76	10.98	12.95	87.73	15.32	11.52	13.15	99.42	0.00	0.00	0.00
TT	95.34	28.98	14.45	19.28	95.35	29.26	14.59	19.47	91.26	28.98	15.59	20.28	99.42	0.00	0.00	0.00
St	95.81	70.34	15.04	24.78	95.81	70.34	15.04	24.78	92.20	70.34	16.23	26.37	99.42	0.00	0.00	0.00
Rul	98.19	72.30	74.26	**73.26**	98.28	74.60	76.61	**75.59**	96.93	72.92	78.05	**75.40**	99.45	54.90	26.42	**35.67**
St/TT	96.20	51.93	29.31	37.48	96.20	52.18	29.45	37.65	92.97	51.93	31.64	39.32	99.42	0.00	0.00	0.00
$St/TT/CF$	94.59	28.71	33.30	30.84	94.61	29.10	33.75	31.25	89.77	28.7	35.94	31.92	99.42	0.00	0.00	0.00
$St/TT/Rul$	98.01	67.52	74.91	71.02	98.11	69.61	77.23	73.23	96.58	67.94	78.76	72.95	99.45	54.90	26.42	**35.67**
all	96.28	46.07	75.02	57.09	96.37	47.50	77.34	58.85	93.11	45.88	78.8	58.01	99.45	54.90	26.42	**35.67**

Table 1: Results achieved by NERU based on the GermEval development set.

4 Evaluation

The evaluation of the progam will be done by the standard precision, recall and F1 score metrcis and some enhanced metrics, which is used to determine the overall ranking of the system. [5]

NERU was evaluated on the GermEval development set. We tested a number of settings: CF – tagging the data only based on the probabilities calculated on the context frequencies, TT – tagging the data only based on TreeTagger's POS tags, St – using only the Stanford NER, Rul – employing only the handcrafted rules. Further, combinations of these settings are also tested. In table 1, we list the respective system scores.

Considering the results on the strict evaluation setting, NER based only on context probabilities (CF) achieves 12.58%, which is the lowest performing setting of the system, followed by the use of the TreeTagger (TT) with 19.28% and the Stanford NER (St) with 24.78%. Surprisingly, NERU's best performance (73.26%) is reached only via the use of handcrafted rules (Rul), where all external tools (TreeTagger and Stanford NER) are not used. Using the information provided by the latter leads to a decrease of system performance to 71.02% ($St/TT/Rul$). This is a somewhat surprising result, considering the fact that the TreeTagger and the Stanford NER identify a significant portion of the NEs on their own (St/TT) reaching a score of 37.48%. Our assumption, however, is that this additional information contradicts the conclusions met by the rules that are solely based on gazetteers and abbreviation lists, which also leads to the decrease of scores. Thus, the final version of the system that we used for the annotation of the GermEval test set employs only the system setting Rul.

Looking deeper into this system setting (based on the system scores presented in table 2), we can see that NERU does not tag at all a large portion of the NE subtypes: *ORGderiv, OTHderiv, OTHpart, PERderiv, PERpart*. After qualitatively evaluating a sample of the system output, we could see that most of these subtypes are generally marked as their supertypes, e.g. *ORGderiv* is tagged as *ORG*. Another observation we could make on this sample is the fact that NERU tends to overgenerate and mark a good portion of non-NE tokens as NEs, e.g. *Bundeswehr*, *Waffen-SS* or *Bundesliga*.

[5] GermEval 2014 NER Evaluation plan `http://is.gd/eval2014`

NE Typ	Precision	Recall	FB1
LOC	84.42%	85.14%	84.78
LOCderiv	88.28%	89.79%	89.03
LOCpart	92.11%	67.31%	77.78
ORG	54.69%	69.15%	61.08
ORGderiv	0.00%	0.00%	0.00
ORGpart	96.55%	92.31%	94.38
OTH	61.27%	57.43%	59.28
OTHderiv	0.00%	0.00%	0.00
OTHpart	0.00%	0.00%	0.00
PER	75.89%	87.41%	81.25
PERderiv	0.00%	0.00%	0.00
PERpart	0.00%	0.00%	0.00
Strict			73.26

Table 2: Detailed scores on the strict evaluation setting based on the Rul system setting.

4.1 Offical Score

Regarding the offical score (Benikova et al., 2014a) NERU lost 25 % of performace in comparison with the development set. The system reached an accuracy of 96.96, a precision of 62.57, a recall of 48.35 and a resulting F_1 of 54.55 in the test set run. The score was calculated by the offical metrics used for the GermEval 2014 Shared Task. An explanation of this losses could be that NERU was also trained with the develop-

Metric	Acc.	P	R	F1
strict	96.96	62.57	48.35	54.55
loose	97.00	63.62	49.16	55.46
outer	94.56	63.69	51.33	56.84
inner	99.37	33.85	12.62	18.39

Table 3: Offical results on test set for all metrics.

ment set in some special cases. Also, as previously mentioned, we did not tag all Named Entity subtypes (6 ouf of 12 types are not taken into consideration).

5 Conclusion

The current paper presents the NER system NERU, which makes use of handcrafted rules, gazetteers and external NER tools for the recognition of NEs in the data. We evaluated the system on the GermEval development set, which showed that the handcrafted rules that do not use the information provided by the TreeTagger and the Stanford NER reach optimal system performance. These rules are solely based on gazetteers and manually created abbreviation lists. Using the latter, NERU participated in the GermEval 2014 NER Shared Task reaching 73.26% on the strict evaluation setting, which is a considerably good performance for German with respect to the scores reported for this language during the CoNLL-2003 Shared Task.

References

Fahd Alotaibi and Mark Lee. 2013. Automatically Developing a Fine-grained Arabic Named Entity Corpus and Gazetteer by utilizing Wikipedia. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 392–400, Nagoya, Japan, October. Asian Federation of Natural Language Processing.

Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Pado. 2014a. Germeval 2014 named entity recognition: Companion paper. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, Hildesheim, Germany.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014b. NoSta-D Named Entity Annotation for German: Guidelines and Dataset. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Jenny Rose Finkel and Christopher D. Manning. 2009. Nested named entity recognition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 141–150, Singapore, August. Association for Computational Linguistics.

Faryal Jahangir, Waqas Anwar, Usama Ijaz Bajwa, and Xuan Wang. 2012. N-gram and Gazetteer List Based Named Entity Recognition for Urdu: A Scarce Resourced Language. In *Proceedings of the 10th Workshop on Asian Language Resources*, pages 95–104, Mumbai, India, December. The COLING 2012 Organizing Committee.

Jun'ichi Kazama and Kentaro Torisawa. 2008. Inducing Gazetteers for Named Entity Recognition by Large-Scale Clustering of Dependency Relations. In *Proceedings of ACL-08: HLT*, pages 407–415, Columbus, Ohio, June. Association for Computational Linguistics.

Sudha Morwal, Nusrat Jahan, and Deepti Chopra. 2012. Named Entity Recognition using Hidden Markov Model (HMM). In *International Journal on Natural Language Computing (IJNLC)*, volume 1.

István Nagy T., Gábor Berend, and Veronika Vincze. 2011. Noun Compound and Named Entity Recognition and their Usability in Keyphrase Extraction. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 162–169, Hissar, Bulgaria, September. RANLP 2011 Organising Committee.

Vassilina Nikoulina, Agnes Sandor, and Marc Dymetman. 2012. Hybrid Adaptation of Named Entity Recognition for Statistical Machine Translation. In *Proceedings of the Second Workshop on Applying Machine Learning Techniques to Optimise the Division of Labour in Hybrid MT*, pages 1–16, Mumbai, India, December. The COLING 2012 Organizing Committee.

Kashif Riaz. 2010. Rule-Based Named Entity Recognition in Urdu. In *Proceedings of the 2010 Named Entities Workshop*, pages 126–135, Uppsala, Sweden, July. Association for Computational Linguistics.

Sujan Kumar Saha, Sudeshna Sarkar, and Pabitra Mitra. 2008. A Hybrid Feature Set based Maximum Entropy Hindi Named Entity Recognition. In *IJCNLP*, pages 343–349.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *International*

Conference on New Methods in Language Processing, pages 44–49, Manchester, UK.

Helmut Schmid. 1999. Improvements in Part-of-Speech Tagging with an Application to German. In Susan Armstrong, Kenneth Church, Pierre Isabelle, Sandra Manzi, Evelyne Tzoukermann, and David Yarowsky, editors, *Natural Language Processing Using Very Large Corpora*, volume 11 of *Text, Speech and Language Processing*, pages 13–26. Kluwer Academic Publishers, Dordrecht.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada.

GESTALT

IGGSA Shared Tasks on German Sentiment Analysis (GESTALT)

Josef Ruppenhofer[‡], **Roman Klinger**[*†], **Julia Maria Struß**[‡],
Jonathan Sonntag[§], **Michael Wiegand**[◦]
[‡] Dept. of Information Science and Language Technology, Hildesheim University
[†] Institute for Natural Language Processing, University of Stuttgart
[*] Semantic Computing Group, CIT-EC, Bielefeld University
[§] Computational Linguistics, Potsdam University
[◦] Spoken Language Systems, Saarland University
{ruppenho,julia.struss}@uni-hildesheim.de
roman.klinger@ims.uni-stuttgart.de
jonathan.sonntag@yahoo.de
michael.wiegand@lsv.uni-saarland.de

Abstract

We present the German Sentiment Analysis Shared Task (GESTALT) which consists of two main tasks: *Source, Subjective Expression and Target Extraction from Political Speeches (STEPS)* and *Subjective Phrase and Aspect Extraction from Product Reviews (StAR)*. Both tasks focused on fine-grained sentiment analysis, extracting aspects and targets with their associated subjective expressions in the German language. STEPS focused on political discussions from a corpus of speeches in the Swiss parliament. StAR fostered the analysis of product reviews as they are available from the website Amazon.de. Each shared task led to one participating submission, providing baselines for future editions of this task and highlighting specific challenges. The shared task homepage can be found at https://sites.google.com/site/iggsasharedtask/.

1 Introduction

In opinion mining, we are not only interested in detecting the presence of opinions (or more broadly, subjectivity) but determining particular attributes. We want to determine *which* valence or polarity an opinion has (positive, negative or neutral), *how* strong it is (intensity), and also know *whose* opinion it is and *what* it is about. The last two questions are what the task of opinion source and target extraction is concerned with. Source and target extraction are capabilities needed for the analysis of unrestricted language texts, where this kind of information cannot be derived from meta-data and where opinions by multiple sources and about multiple, potentially related, targets appear side by side.

We present two shared tasks that ran under the auspices of the Interest Group of German Sentiment Analysis[1] (IGGSA). Maintask 1 on *Source, Subjective Expression and Target Extraction from Political Speeches (STEPS)* constitutes the first evaluation campaign for source and target extraction on German language data. Maintask 2 on *Subjective Phrase and Aspect Extraction from Product Reviews (StAR)* focuses on the aspect extraction, which is understood as the target of a subjective phrase. For both tasks, publicly available resources have been created, which serve as a reference corpus for the evaluation of opinion source and target extraction in German.

2 Task Descriptions

In this section, we present the task setting, describe the dataset, the annotation, the subtasks, the evaluation and results for each of the two main tasks (Section 2.1 and Section 2.2), respectively.

2.1 Maintask 1

Maintask 1 calls for the identification of subjective expressions, sources and targets in parliamentary speeches. While these texts can be expected to be opinionated, they pose the challenges that

[1] https://sites.google.com/site/iggsahome/

sources other than the speaker may be relevant and that the targets, though constrained by topic, can vary widely. As in the case of Maintask 2, the dataset provided is the first one that provides publicly available expression-level annotations on running texts of this type for German.

2.1.1 Dataset

The STEPS data set stems from the debates of the Swiss parliament (*Schweizer Bundesversammlung*).[2] This particular data set was selected for two reasons. First, the source data is open to the public and we can re-distribute it with our annotations. We were not able to fully ascertain the copyright situation for German parliamentary speeches, which we had also considered. Second, the text calls for annotation of multiple sources and targets.

As the Swiss parliament is a multi-lingual institution, we were careful to exclude not only non-German speeches but also German speeches that constitute responses to, or comments on, speeches, heckling, and side questions in other languages. This way, our annotators did not have to label any German data whose correct understanding might rely on material in a language that they might not be able to interpret correctly.

Some potential linguistic difficulties consisted in peculiarities of Swiss German found in the data. For instance, the vocabulary of Swiss German is different from standard German, often in subtle ways. For instance, the verb *vorprellen* is used in the following example instead of *vorpreschen*, which would be expected for German spoken in Germany:

> *Es ist unglaublich: Weil die Aussenministerin vorgeprellt ist, kann man das nicht mehr zurücknehmen. (Hans Fehr, Frühjahrsession 2008, Zweite Sitzung – 04.03.2008)*[3]
>
> 'It is incredible: because the foreign secretary acted rashly, we cannot take that back again.'

In order to reduce any negative impact that might come from misreadings of the Swiss German by our annotators, who were German and Austrian rather than Swiss, we selected speeches about what we deemed to be non-parochial issues. For instance, we picked texts on international affairs rather than ones about Swiss municipal governance.

Technically, the STEPS data underwent the following pre-processing pipeline. Sentence segmentation and tokenization was done using OpenNLP[4], followed by lemmatization with the TreeTagger (Schmid, 1994), constituency parsing by the Berkeley parser (Petrov and Klein, 2007), and final conversion of the parse trees into TigerXML-Format using TIGER-tools (Lezius, 2002). To perform the annotation we used the Salto-Tool (Burchardt et al., 2006).

2.1.2 Annotation

Through our annotation scheme[5], we provide annotations at the expression level. No sentence or document-level annotations are manually performed or automatically derived.

There were no restrictions imposed on annotations. The subjective expressions could be verbs, nouns, adjectives or multi-words. The sources and targets could refer to any actor or issue as we did not focus on anything in particular.

The definition of subjective expressions (SE) that we used is broad and based on well-known prototypes. It largely follows the model of what Wilson and Wiebe (2005) subsume under the umbrella term *private state*, as defined by Quirk et al. (1985): "As a result, the annotation scheme is centered on the notion of private state, a general term that covers opinions, beliefs, thoughts, feelings, emotions, goals, evaluations, and judgments.":

- evaluation (positive or negative): *toll* 'great', *doof* 'stupid'

[2] The full task test data is available at `https://sites.google.com/site/iggsasharedtask/home/testdata-maintask1-salto_tiger-xml.zip` . The subtask test data for is at `https://sites.google.com/site/iggsasharedtask/home/testdata-maintask1-subtasks-salto_tiger.xml.zip`.

[3] `http://www.parlament.ch/ab/frameset/d/n/4802/263473/d_n_4802_263473_263632.htm`

[4] `http://opennlp.apache.org/`

[5] See `https://sites.google.com/site/iggsasharedtask/task-1/STEPS_guide.pdf` for the the guidelines we used.

Name	Source		Target		Frame	
SwissGerman	*not applicable*				14	
RhetoricalDevices	*not applicable*				64	
Inferred	344	(7.8%)	177	(3.9%)	97	(2.0%)
Uncertain	61	(1.4%)	29	(0.6%)	58	(1.2%)

Table 1: Flags annotated across all annotators and files of Maintask 1

	F_1	Dice for true positives
Subjective Expression	63.32	0.92
Sources*	68.70	0.99
Targets*	80.63	0.85

Table 2: Average inter-annotator agreement across all pairs of annotators on test data of Maintask 1 (F_1 is based on partial overlap; Dice quantifies the amount of overlap for matches)

- (un)certainty:
  zweifeln 'doubt', *gewiss* 'certain'
- emphasis:
  sicherlich/bestimmt 'certainly'
- speech acts:
  sagen 'say', *ankündigen* 'announce'
- mental processes:
  denken 'think', *glauben* 'believe'

Beyond giving the prototypes, we did not seek to impose on our annotators any particular definition of subjective or opinion expressions from the linguistic, natural language processing or psychological literature related to subjectivity, appraisal, emotion or related notions.

In marking subjective expressions, the annotators were told to select minimal spans. This guidance was given because we had decided that within the scope of this shared task we would forgo any treatment of polarity and intensity. Accordingly, negation, intensifiers and attenuators and any other expressions that might affect a minimal expression's polarity or intensity could be ignored.

When labeling sources and targets, annotators were asked to first consider syntactic and semantic dependents of the subjective expressions. If sources and targets were locally unrealized, the annotators could annotate other phrases in the context. Where a subjective expression represented the view of the implicit speaker or text author, annotators could indicate this by setting a flag *Sprecher* 'Speaker' on the the source element.

For all three types of labels, subjective expressions, sources, and targets, annotators had the option of using two additional flags. The first flag was intended to mark a label instance as *Inferiert* 'Inferred'. In the case of subjective expressions, this covers, for instance, cases where annotators were not sure if an expression constituted a polar fact or an inherently subjective expression. In the case of sources and targets, the 'inferred' label applies to cases where the referents cannot be annotated as local dependents but have to be found in the context. The second flag afforded annotators the ability to mark an annotation as *Unsicher* 'Uncertain', if they were unsure whether the span should really be labeled with the relevant category.

The annotators were asked to use a flag *Rhetorisches Stilmittel* 'Rhetorical device' for subjective expression instances where subjectivity was conveyed through some kind of rhetorical device such as repetition. Across all three annnotators, 64 instances were labeled as 'rhetorical de-

Run	Measure	Subjective Expression	Source	Source_SE	Target	Target_SE
Run 3	Prec	63.42	**48.55**	**74.89**	**56.25**	79.71
	Rec	26.10	11.32	**42.46**	15.60	**58.00**
	F_1	36.98	18.36	**54.19**	24.43	**67.14**
Run 5	Prec	**80.56**	47.98	58.55	*not applicable*	
	Rec	29.97	10.44	32.65	*not applicable*	
	F_1	43.69	17.14	41.92	*not applicable*	

Table 3: Best participant runs for Maintask 1 (3 = rule-based system; 5 = translation-based system, which did not include Targer identification. Results suffixed with subjective expressions consider only cases where the system already matched the gold standard on the subjective expression)

vice' in the data.

Finally, the annotation guidelines gave annotators the option to mark particular subjective expressions as *Schweizerdeutsch* 'Swiss German' when they involved language usage that they were not fully familiar with. Such cases could then be excluded or weighted differently for the purposes of system evaluation. In our annotation, these markings were in fact rare with only 14 of such flag instances across all three annotators.

Summing over all three annotators, our dataset covers 1815 sentences. In total, 4935 subjective expression frame instances were labeled by the annotators combined (2.7 frames/sentence). Related to the frames, 8959 frame element (source or target) instances were annotated (1.8 frame elements/frame). Although the theory embodied by our guidelines calls for at least one source and target label per annotated subjective expression frame, we find slightly less than one instance of each (4427 sources, 4532 targets). In Table 1, we see that not many flags were annotated by our annotators. The careful selection of our data with respect to the topics treated seems to have worked well. We have few instances of subjective expressions that were flagged as Swiss German formulations by our annotators. The most common type of flag was the one for 'inferred' labels. Here, inference of sources was by far the most common case. Note, that fewer labels were marked 'uncertain' than were marked 'inferred'. Inference did not necessarily result in uncertainty.

In Table 2, we present results on the inter-annotator agreement on the test data. One way of measuring the agreement uses the precision/recall-framework of evaluation. We calculate the relevant numbers based on treating one annotator as gold and another as system, and averaging the results for the three pairs of annotators. For F_1, we counted a true positive when there was partial span overlap. In addition, we present a token-based multi-κ value (Davies and Fleiss, 1982). Given that in our annotation scheme, a single token can be e.g. a target of one subjective expression while itself being a subjective expression as well, we need to calculate three kappa values covering the binary distinctions between presence of each label and its absence. For subjective expressions κ is 0.39, for sources 0.57, and for targets 0.46.

As exact matches on spans are relatively rare, the Dice coefficient is used to measure the overlap between a system annotation and a gold standard annotation (Dice, 1945). The Dice coefficient $\mathrm{dc}(S, G)$ is a similarity measure ranging from 0 to 1, where

$$\mathrm{dc}(S, G) = \frac{2|S \cap G|}{|S| + |G|},$$

and G is the set of tokens in the gold annotations and S the set of tokens the prediction (the system label), respectively.

2.1.3 Subtasks

The STEPS shared task offered a full task as well as two subtasks:

Full task Identification of subjective expressions with their respective sources and targets.

Subtask 1 Participants are given the subjective expressions and are only asked to identify opinion sources.

Subtask 2 Participants are given the subjective expressions and are only asked to identify opinion targets.

Participants could choose any combination of the tasks. However, so as to not give an unfair advantage, the full task was run and evaluated before the gold information on subjective expressions was given out for the two subtasks, which were run concurrently.

2.1.4 Evaluation Metrics

The runs that were submitted by the participants of the shared task were evaluated on different levels, according to the task they chose to participate in. For the full task, there was an evaluation of the subjective expressions as well as the targets and sources for subjective expressions, matching the system's annotations against those in the gold standard. For subtasks 1 and 2, only the sources and targets were evaluated, as the subjective expressions were already given.

In this first iteration of the STEPS task, we evaluated against each of our three annotators individually rather than against a single gold-standard. Our intent behind this choice was to retain the variation between the annotators.

We used recall to measure the proportion of correct system annotations with respect to the gold standard annotations. Additionally, precision was calculated so as to give the fraction of correct system annotations relative to all the system annotations. As we did for inter-annotator-agreement, for recall and precision we counted a match when there was partial span overlap. Similarly, we again used the Dice coefficient to assess the overlap between a system annotation and a gold standard annotation.

The group that participated in our main task submitted five different runs, based on two different system architectures. Table 3 shows the best result for each architecture. The scores represent averages across the comparisons relative to each of the three annotators. The rule-based system generally performed better than the translation-based one. However, the latter was much better in its precision on recognizing subjective expressions in the full task. As is to be expected, when the system had already matched the gold standard on the subjective expressions, its performance on source and target recognition, shown in columns Source_SE , Target_SE, is much superior to performance in the general case.

2.2 Maintask 2: Subjective Phrase and Aspect Extraction from Product Reviews

Maintask 2 was designed to foster the development of systems to automatically extract subjective, evaluative phrases from German Amazon reviews, aspects described in the review and their relation, i.e., which evaluative phrase targets which aspect. In addition, another focus is cross-domain learning: The development corpus consists of reviews for various products while the test corpus is from yet another product not known to the participants before.

2.2.1 Dataset

For this task, a data set was provided for training parameters and developing the system. *The USAGE Review Corpus for Fine Grained Multi Lingual Opinion Analysis* (Klinger and Cimiano, 2014) was previously published and was fully available to the participants from the start of the task on. It consists of 611 German and 622 English reviews for coffee machines, cutlery sets, microwaves, toasters, trashcans, vacuum cleaners, and washers from which only the German part has been used in this shared task. To construct the test corpus, 1646 reviews for the search term *Wasserkocher* 'water boiler' were retrieved. From these, 100 sampled reviews were annotated and included in the test corpus. The training[6] and test[7] data is freely available.

2.2.2 Annotation

The entity classes *aspect* and *evaluative (subjective) expression* are annotated in the corpus. Evaluative expressions are assigned a polarity (posi-

[6]Maintask 2 training data: `http://dx.doi.org/10.4119/unibi/citec.2014.14`

[7]Maintask 2 test data: `http://dx.doi.org/10.4119/unibi/2695161`

tive, negative, neutral), which is not used in this shared task, and a set of aspects they refer to. The annotators were instructed to regard everything as an aspect that is part of a product or related to it and can influence the opinion about it, including the whole product itself. Evaluative phrases express an opinion. Negations are not separately annotated but are part of a phrase. Annotators were asked to avoid overlapping annotations if possible. The annotations should be as short as possible, as long as the meaning is understandable if only the annotations were given (without the sentence itself).

Every review in the training data is annotated by two linguists, the test data is annotated by one (the information which of the training data annotation corresponds to the annotator of the test data is available).

In the following examples, aspects are marked in blue and subjective phrases are marked in red:

Ich hatte keine Probleme mit der Rückgabe .
I had no problems with the return .

return is a target of *no problems*.
no problems is positive.

Die Waschmaschine selbst ist toll , der beiliegende Schlauch ist Schrott.
The washer itself is great , the included hose is junk .

washer is a target of *great*.
hose is a target of *junk*.
great is positive.
junk is negative.

Es sieht sehr hübsch aus, wie ein Aufbewahrungsbehälter , er ist leicht und einfach zu benutzen .
It looks very neat , like a storage container , and using it is very simple and easy .

– *looks* is a target of *very neat*.
using is a target of *simple* and of *easy*.

The inter-annotator agreement of the full training corpus is $\kappa = 0.65$ (Cohen's κ). The inter-annotator F_1 measure is 0.71 for aspects, 0.55 for subjective phrases and 0.42 for the relations between both (including an error propagation of having the exact same phrases annotated). These measures can be regarded as upper bounds for meaningful results of an automated approach.

Table 4 presents the main statistics of the training and testing corpora. Here, annotator 1 of the training corpus performed the annotation of the test data. Obviously, the number of annotated phrases is higher in the test data.

The most frequent subjective phrases for the different products are very similar. For instance, the phrases *gut* 'good' and *sehr zufrieden* 'very satisfied' occurs in all top 10 lists of subjective phrases. However, the most frequent aspect phrases are very different, as the product category itself is frequently used as an aspect (e.g. *Kaffeemaschine* 'coffee maker' or *Besteck* 'cutlery'). In addition, very product class-specific aspects are mentioned frequently, like *Wasser* 'water', *schneiden* 'cut', or *Edelstahl* 'stainless steel'. Some aspects are shared between product categories, for instance *Preis* 'price' or *Qualität* 'quality'.

Clearly, the cross-domain inference task is more challenging, as the mentioned aspects are not as similar as the annotated subjective phrases.

2.2.3 Subtasks

The three substasks to be addressed by the participants were:

Subtask 2a Identication of subjective phrases.

Subtask 2b Identification of aspect phrases.

Subtask 2c Identification of subjective phrases and aspect phrases and indication for each aspect phrase of which subjective phrase it is the target (if any).

2.2.4 Evaluation metrics and Baseline approach

For evaluation, the F_1 measure of the exact match of the predicted phrases in comparison to the annotated phrases is taken into account. This is straight-forward for Subtasks 2a and 2b. In 2c, a pair of aspect and subjective phrase was considered to be correctly identified, if both phrases

	Train Ann. 1	Train Ann. 2	Test
Number of reviews	611		100
Number of products	127		100
Number of Aspects	6340	5055	1662
Number of Aspects/Review	10.4	8.3	16.6
Number of positive Subj.	3840	3717	823
Number of positive Subj./Review	6.3	6.1	8.2
Number of negative Subj.	1094	1052	264
Number of negative Subj./Review	1.8	1.7	2.6
Target Rel.	4085	4643	1013
Target Rel./Review	6.7	7.6	10.1

Table 4: Statistics of the corpora used in Maintask 2

predicted to be participating were identified correctly (on the phrase level) as well as annotated as a pair.

For comparison, as a baseline, a machine learning-based system optimized for in-domain inference was applied[8] (Klinger and Cimiano, 2013a; Klinger and Cimiano, 2013b). A comparison of the participant's result and the baseline is shown in Table 5. It can be observed that the baseline outperforms the subjective phrase detection, but the result submitted by the participant is superior in the more difficult cross-domain tasks of aspect extraction. The extraction of relations clearly remains a challenge.

3 Related Work

While quite a few shared tasks have addressed the recognition of subjective units of language and, possibly, the classification of their polarity (SemEval 2013 Task 2, Twitter Sentiment Analysis (Nakov et al., 2013); SemEval-2010 task 18: Disambiguating sentiment ambiguous adjectives (Wu and Jin, 2010); SemEval-2007 Task 14: Affective Text (Strapparava and Mihalcea, 2007) *inter alia*), few tasks have included the extraction of sources and targets.

The prior work most relevant to the tasks presented here was done in the context of the Japanese NTCIR[9] Project. In the NTCIR-6 Opinion Analysis Pilot Task (Seki et al., 2007), which was offered for Chinese, Japanese and English, sources and targets had to be found relative to whole opinionated sentences rather than individual subjective expressions. However, the task allowed for multiple opinion sources to be recorded for a given sentence if there were multiple expressions of opinion. The opinion source for a sentence could occur anywhere in the document. In the evaluation, as necessary, co-reference information was used to (manually) check whether a system response was part of the correct chain of co-referring mentions. The sentences in the document were judged as either relevant or non-relevant to the topic (=target). Polarity was determined at the sentence level. For sentences with more than one opinion expressed, the polarity of the main opinion was carried over to the sentence as a whole. All sentences were annotated by three raters, allowing for strict and lenient (by majority vote) evaluation. The subsequent Multilingual Opinion Analysis tasks NTCIR-7 (Seki et al., 2008) and NTCIR-8 (Seki et al., 2010) were basically similar in their setup to NTCIR-6.

While GESTALT shared tasks focussed on German, the most important difference to the shared tasks organized by NTCIR is that it defined the source and target extraction task at the level of individual subjective expressions. There was no comparable shared task annotating at the expression level, rendering existing guidelines imprac-

[8] A high-recall combination of the joint configuration and the pipeline setting has been applied.

[9] NII [National Institute of Informatics] Test Collection for IR Systems

	Baseline			Participant		
Subtask	Prec	Rec	F_1	Prec	Rec	F_1
Aspect Phrase	**65.5**	46.4	54.3	55.5	62.2	**58.7**
Subjective Phrase	51.5	**41.4**	**45.9**	**51.6**	32.0	39.5
Relation	**15.9**	8.3	10.9	12.6	**13.8**	**13.2**

Table 5: Results of the baseline system and the participant's best submission in Maintask 2.

tical and necessitating the development of completely new guidelines.

Another more recent shared task related to GESTALT is the Sentiment Slot Filling track (SSF) that was part of the Shared Task for Knowledge Base Population of the Text Analysis Conference (TAC) organised by the National Institute of Standards and Technology (NIST) (Mitchell, 2013). The major distinguishing characteristic of that shared task, which is offered exclusively for English language data, lies in its retrieval-like setup. Here, the task is to extract all possible opinion sources and targets from a given text. By contrast, in SSF the task is to retrieve sources that have some opinion towards a given target entity or targets of some given opinion sources. In both cases, the polarity of the underlying opinion is also specified within SSF. The given targets or sources are considered a type of *query*. The opinion sources and targets are to be retrieved from a document collection.[10] Unlike GESTALT, SSF uses heterogeneous text documents including both newswire and discussion forum data from the Web.

This year's SemEval-2014 Task 4 on Aspect Based Sentiment Analysis (ABSA) on English review data for restaurant and laptop reviews (Pontiki et al., 2014) constitutes another related shared task. It focused on aspect-based polarity detection. The main differences are that the aspect categories were predefined and that the polarity assignment did not include the detection of the evaluative phrases. Therefore, the polarity assignment was on the aspect level and the relation between a subjectivity-bearing word was implicit. Another difference between ABSA and GESTALT (StAR, specifically) is that the number of products taken into account is higher in StAR, motivating a cross-domain inference challenge.

4 Conclusion and Outlook

We reported on the first iteration of two shared tasks for German sentiment analysis. Both tasks focused on the discovery of subjective expressions and their related entities. In the case of STEPS, sources and targets had to be found and linked to subjective expressions in political speeches, in the case of StAR, aspects had to be identified and tied to subjective expressions in Amazon reviews.

Although a preliminary call for interest had indicated interest by 3–4 groups for each of the tasks, in the end each task had only one participant. We therefore solicited feedback from actual and potential participants at the end of the IGGSA-GESTALT workshop in order to be able to tailor the tasks better in a future iteration.

Based on the discussion, both shared tasks plan on including polarity in the evaluation for their next iteration. For both tasks, there was discussion what a suitable evaluation procedure would be, in particular whether partial matches should be the basis of the main measures or if exact matches would be more desirable.

Specific to STEPS, we are considering conducting the evaluation in alternative ways on a future iteration of the task. One direction to pursue is to derive new versions of the gold standard based on the level of inter-annotator agreement on the labels. In a full-agreement mode, we would only retain annotations of the gold standard that had majority or even full agreement on the subjective expression level for all three annotators. Another alternative would consist in establishing an expert-adjudicated gold-standard, after all. The benefit of any of these alterna-

[10] In 2014, the text from which entities are to be retrieved is restricted to one document per query.

tive evaluation modes would be that a clear objective function can be learnt and that the upper bound for system performance would again be 100% precision/recall/F_1-score, whereas it was lower for this iteration given that existing differences between the annotators necessarily led to false positives and negatives.

For the next iteration of GESTALT, we plan to make a baseline system available, such that the barrier to participation in the shared task is lower and participants' efforts can be focused on the actual methods.

Acknowledgments

We would like to thank Simon Clematide for helping us get access to the Swiss data for the STEPS task. For their support in preparing and carrying out the annotations of this data, we would like to thank Jasper Brandes, Melanie Dick, Inga Hannemann, and Daniela Schneevogt. We thank the German Society for Computational Linguistics for its financial support of the STEPS annotation effort. For the annotations used in the StAR task, we thank Luci Fillinger and Frederike Strunz. Roman Klinger was partially funded by the *It's OWL* project ('Intelligent Technical Systems Ostwestfalen-Lippe', `http://www.its-owl.de/`), a leading-edge technology and research cluster funded by German Ministry of Education and Research (BMBF). This first and last author were partially supported by the German Research Foundation (DFG) under grants RU 1873/2-1 and WI 4204/2-1, respectively.

References

Aljoscha Burchardt, Katrin Erk, Anette Frank, Andrea Kowalski, and Sebastian Pado. 2006. SALTO - A Versatile Multi-Level Annotation Tool. In *Proceedings of the 5th Conference on Language Resources and Evaluation*, pages 517–520.

Mark Davies and Joseph L. Fleiss. 1982. Measuring agreement for multinomial data. *Biometrics*, 38(4):1047–1051.

Lee R. Dice. 1945. Measures of the Amount of Ecologic Association Between Species. *Ecology*, 26(3):297–302.

Roman Klinger and Philipp Cimiano. 2013a. Bidirectional inter-dependencies of subjective expressions and targets and their value for a joint model. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 848–854, Sofia, Bulgaria, August. Association for Computational Linguistics.

Roman Klinger and Philipp Cimiano. 2013b. Joint and pipeline probabilistic models for fine-grained sentiment analysis: Extracting aspects, subjective phrases and their relations. In *Data Mining Workshops (ICDMW), 2013 IEEE 13th International Conference on*, pages 937–944, Dec.

Roman Klinger and Philipp Cimiano. 2014. The usage review corpus for fine grained multi lingual opinion analysis. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Wolfgang Lezius. 2002. TIGERsearch - Ein Suchwerkzeug für Baumbanken. In Stephan Busemann, editor, *Proceedings of KONVENS 2002*, Saarbrücken, Germany.

Margaret Mitchell. 2013. Overview of the TAC2013 Knowledge Base Population Evaluation: English Sentiment Slot Filling. In *Proceedings of the Text Analysis Conference (TAC)*, Gaithersburg, MD, USA.

Preslav Nakov, Sara Rosenthal, Zornitsa Kozareva, Veselin Stoyanov, Alan Ritter, and Theresa Wilson. 2013. SemEval-2013 Task 2: Sentiment Analysis in Twitter. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 312–320, Atlanta and Georgia and USA. Association for Computational Linguistics.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 404–411, Rochester, New York, April. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Randolph Quirk, Sidney Greenbaum, Geoffry Leech, and Jan Svartvik. 1985. *A comprehensive grammar of the English language*. Longman.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, Manchester, UK.

Yohei Seki, David Kirk Evans, Lun-Wei Ku, Hsin-Hsi. Chen, Noriko Kando, and Chin-Yew Lin. 2007. Overview of opinion analysis pilot task at ntcir-6. In *Proceedings of NTCIR-6 Workshop Meeting*, pages 265–278.

Yohei Seki, David Kirk Evans, Lun-Wei Ku, Le Sun, Hsin-Hsi Chen, and Noriko Kando. 2008. Overview of multilingual opinion analysis task at NTCIR-7. In *Proceedings of the 7th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering, and Cross-Lingual Information Access*, pages 185–203.

Yohei Seki, Lun-Wei Ku, Le Sun, Hsin-Hsi Chen, and Noriko Kando. 2010. Overview of Multilingual Opinion Analysis Task at NTCIR-8: A Step Toward Cross Lingual Opinion Analysis. In *Proceedings of the 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access*, pages 209–220.

Carlo Strapparava and Rada Mihalcea. 2007. SemEval-2007 Task 14: Affective Text. In Eneko Agirre, Lluís Màrquez, and Richard Wicentowski, editors, *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 70–74. Association for Computational Linguistics.

Theresa Wilson and Janyce Wiebe. 2005. Annotating attributions and private states. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pages 53–60, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Yunfang Wu and Peng Jin. 2010. SemEval-2010 Task 18: Disambiguating Sentiment Ambiguous Adjectives. In Katrin Erk and Carlo Strapparava, editors, *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 81–85, Stroudsburg and PA and USA. Association for Computational Linguistics.

Saarland University's Participation in the GErman SenTiment AnaLysis shared Task (GESTALT)

Michael Wiegand, Christine Bocionek, Andreas Conrad, Julia Dembowski
Jörn Giesen, Gregor Linn, Lennart Schmeling
Spoken Language Systems
Saarland University
D-66123, Saarbrücken, Germany
michael.wiegand@lsv.uni-saarland.de

Abstract

We report on the two systems we built for Task 1 of the German Sentiment Analysis Shared Task, the task on *Source, Subjective Expression and Target Extraction from Political Speeches (STEPS)*. The first system is a rule-based system relying on a predicate lexicon specifying extraction rules for verbs, nouns and adjectives, while the second is a translation-based system that has been obtained with the help of the (English) MPQA corpus.

1 Introduction

In this paper, we describe our two systems for Task 1 of the German Sentiment Analysis Shared Task, the task on *Source, Subjective Expression and Target Extraction from Political Speeches (STEPS)* (Ruppenhofer et al., 2014). In that task, both *opinion sources*, i.e. the entities that utter an opinion, and *opinion targets*, i.e. the entities towards which an opinion is directed, are extracted from German sentences. The opinions themselves have also to be detected automatically. The sentences originate from debates of the Swiss Parliament (*Schweizer Bundesversammlung*).

The first system is a rule-based system relying on a predicate lexicon specifying extraction rules for verbs, nouns and adjectives, while the second is a translation-based system that has been obtained with the help of the (English) MPQA corpus (Wiebe et al., 2005).

This shared task has been organized for the first time. No labeled training data have been available.

2 Rule-based System

The pipeline of the rule-based system is displayed in Figure 1. The major assumption that underlies this system is that the concrete realization of opinion sources and targets is largely determined by the opinion predicate[1] by which they are evoked. Therefore, the task of extracting opinion sources and targets is a lexical problem, and a lexicon for opinion predicates specifying the argument position of sources and targets is required. For instance, in Sentence (1), the sentiment is evoked by the predicate *liebt*, the source is realized by its subject *Peter* while the target is realized by its accusative object *Maria*.

(1) $[\text{Peter}]^{source}_{subj}$ **liebt**$_{sentiment}$ $[\text{Maria}]^{target}_{obja}$.
(Peter loves Maria.)

With this assumption, we can specify the demands of an opinion source/target extraction system. It should be a tool that given a lexicon with argument information about sources and targets for each opinion predicate

- checks each sentence for the presence of such opinion predicates,
- syntactically analyzes each sentence and
- determines whether constituents fulfilling the respective argument information about

[1] We currently consider verbs, nouns and adjectives as potential opinion predicates.

sources and targets are present in the sentence.

In the following, we describe how we implemented these different steps. The rule-based system will be made publicly available allowing researchers to test different sentiment lexicons with different argument information about opinion sources and targets.[2]

2.1 Linguistic Processing

Even though the data for this task already come in a parsed format, we felt the need to add further linguistic information. In addition to the existing constituency parse provided by the Berkeley parser (Petrov et al., 2006), we also included dependency parse information. With that representation, relationships between opinion predicates and their sources and targets can be formulated more intuitively.[3]

As a dependency parser, we chose *ParZu* (Sennrich et al., 2009). We also carried out some normalization on the parse output in order to have a more compact representation. To a large extent, the type of normalization we carry out is in line with the output of dependency parsers for English, such as the Stanford parser (de Marneffe et al., 2006). It is included since it largely facilitates writing extraction rules. The normalization includes

(a) active-passive normalization

(b) conflating several multi-edge relationships to one-edge relationships

(c) particle-verb reconstruction

Our extraction rules assume a sentence in active voice, therefore sentences in passive voice (we exclusively consider the frequent German *von-Passiv*) need to be converted to active voice (a).[4] This conversion is illustrated in Figure 2.

For our extraction rules, we want to specify the relationship between opinion predicates and their sources/targets as direct (or first-order) dependency relationships. In current dependency parsers for German, however, those two types of entities are often not connected via a direct edge, i.e. they are multi-edge (or second-order) relationships. We, therefore, wrote a set of rules collapsing those multi-edge relationships. A simple example is illustrated in Figure 3 for the case of predicate adjectives and their subjects. In Figure 3(a) *schön* and *Auto* are connected via *pred+subj* which we collapse to just *subj* in Figure 3(b).[5] In a similar fashion, we also collapse prepositional objects as illustrated in Figure 4.

Finally, a considerable fraction of German verbs are *particle verbs* which means that several inflectional forms are split into two tokens, i.e. verb stem and some particle. These two tokens may then be separated by other constituents in a sentence. This is illustrated for *aufgeben* in Sentence (2) which is split in *gab* and *auf*. The ParZu dependency parser connects stems and particles via a dedicated relation edge. Thus the full lemma (as listed in the lexicon specifying the extraction rules) can be reconstructed.

(2) Er *gab* das Rauchen vor 10 Jahren *auf*.
(He *gave up* smoking 10 years ago.)

2.2 The Extraction Rules

As already indicated above, the heart of the rule-based system is a lexicon that specifies the (possible) argument positions of sources and targets. So far, there does not exist a lexicon with that specific information which is why we came up with a set of default rules for the different parts of speech. The set of opinion predicates are the sentiment expressions from the PolArt system (Klenner et al., 2009). (For some runs for the benchmark, we also add sentiment expressions from SentiWS (Remus et al., 2010).) Every mention of such expressions will be considered as a mention of an opinion predicate, that is, we do not carry out any subjectivity word-sense disambiguation (Akkaya et al., 2009).

[2]The code will be made available via the website of the shared task `https://sites.google.com/site/iggsasharedtask/task-1`

[3]As a matter of fact, the most appropriate representation for that task is semantic-role labeling (Ruppenhofer et al., 2008; Kim and Hovy, 2006; Wiegand and Klakow, 2012), however, there currently do not exist any robust tools of that kind for German.

[4]From a semantic point of view, the content of a sentence in passive voice and that of a sentence in active voice are, more or less, identical. Therefore, normalizing passive voice sentences to active voice sentences is legitimate.

[5]The copula *ist* needs to be inserted for syntactic reasons in that sentence. It does not carry any semantic content and, therefore, can be dropped for our purposes.

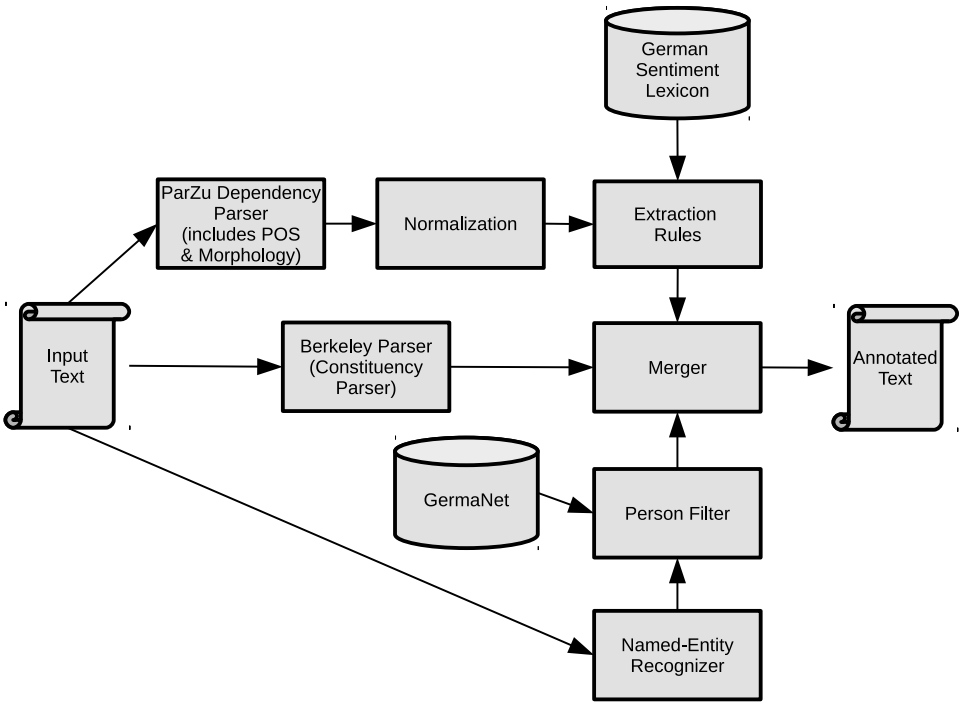
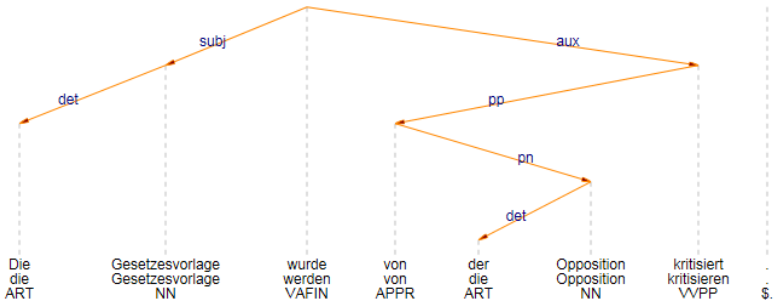
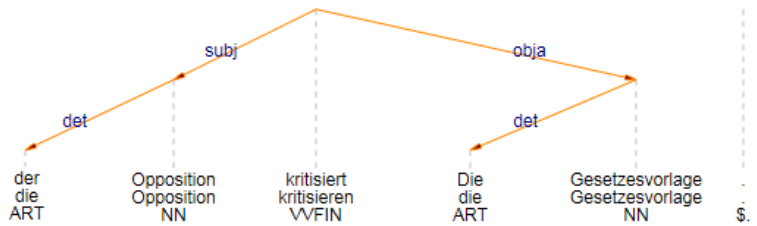


Figure 1: Processing pipeline of the rule-based system.

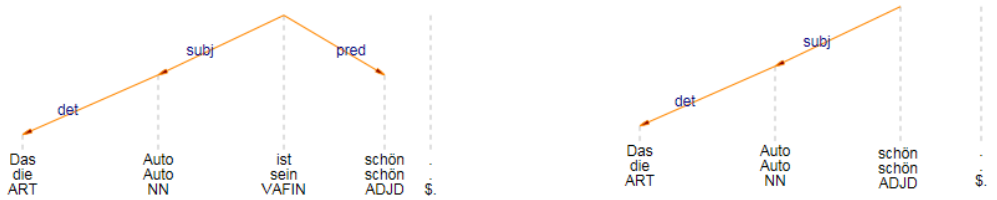


(a) original dependency parse



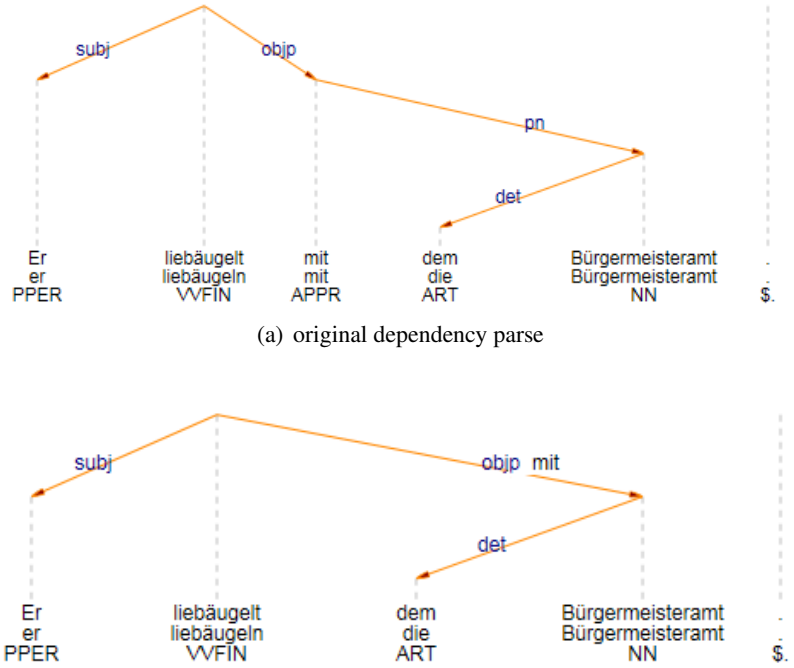
(b) normalized dependency parse

Figure 2: Illustration of normalizing dependency parses with passive voice constructions.



(a) original dependency parse (b) normalized dependency parse

Figure 3: Illustration of normalizing dependency parses with predicative adjectives.



(a) original dependency parse

(b) normalized dependency parse

Figure 4: Illustration of normalizing dependency parses with prepositional complements.

These default extraction rules are designed in such a way that for a large fraction of opinion predicates with the pertaining part of speech they are correct. The rules are illustrated in Table 1. We currently have distinct rules for verbs, nouns and adjectives. All rules have in common that for every opinion predicate mention, at most one source and at most one target is assigned. The rules mostly adhere to the dependency relation labels of ParZu.[6]

The rule for verbs assumes sources in subject and targets in object position (1). Note that for targets, we specify a priority list. That is, the most preferred argument position is an dative object (*objd*), the second most preferred position is an accusative object (*objа*), etc. In computational terms, this means that the classifier checks the entire priority list (from left to right) until a relation has matched in the sentence to be classified. For prepositional complements, we also allow a wildcard symbol (*pobj-**) that matches all prepositional complements irrespective of its particular head, e.g. *über das Freihandelsabkommen* (*pobj-ueber*) in (3).

(3) [Deutschland und die USA]$^{source}_{subj}$ **streiten** [über das Freihandelsabkommen]$^{target}_{pobj-ueber}$.
(Germany and the USA quarrel over the free trade agreement.)

For nouns, we allow determiners (possessives) (4) and genitive modifiers (5) as opinion sources whereas targets are considered to occur as prepositional objects.

(4) [Sein]$^{source}_{det}$ **Hass** [auf die Regierung]$^{target}_{pobj-auf}$ ...
(His hatred towards the government ...)

(5) Die **Haltung** [der Kanzlerin]$^{source}_{gmod}$ [zur Energiewende]$^{target}_{pobj-zu}$ ...
(The chancellor's attitude towards the energy revolution ...)

The rule for adjectives is different from the others since it assumes the source of the adjective to be the speaker of the utterance. Only the target has a surface realization. Either it is an attributive adjective (6) or it is the subject of a predicative adjective (7).

Part of Speech	Source	Target
verb	subj	objd, obja, objc, objі, s, objp-*
noun	det, gmod	objp-*
adjective	*author*	attr-rev, subj

Table 1: Extraction rules for verb, noun and adjective opinion predicates.

(6) Das ist ein [guter]$^{target}_{attr-rev}$ **Vorschlag**.
(This is a good proposal.)

(7) [Der Vorschlag]$^{target}_{subj}$ ist **gut**.
(The proposal is good.)

Our rule-based system is designed in such a way that, in principle, it would also allow more than one opinion frame to be evoked by the same opinion predicate. For example, in *Peter überzeugt Maria/Peter convinces Maria*, one frame sees *Peter* as source and *Maria* as target, and another frame where the roles are switched. Our default rules do not include such cases, since such property is specific to particular opinion predicates.

2.3 Filtering

Our extraction lexicon tends to overgenerate in several situations. This can be mainly ascribed to the fact that we do not carry out any word-sense disambiguation and we use simple default rules. The only means to rectify this shortcoming (to a certain extent) is by applying a heuristic filter. The filter that we apply concerns the plausibility of opinion sources. We only mark a phrase as an opinion source, if it denotes a person or a group of persons. We automatically detect this semantic information with the help of a named-entity recognizer (Faruqui and Padó, 2010) (in order to detect proper nouns) and GermaNet (Hamp and Feldweg, 1997), the German version of WordNet (Miller et al., 1990) (in order to cope with common nouns). In addition, we also formulate a set of rules for personal pronouns, e.g. the German pronoun *es*, similar to the English *it*, is fairly unlikely to denote a human being and therefore is not eligible to represent opinion sources.

[6]The definition of those dependency labels is available at `https://github.com/rsennrich/ParZu/blob/master/LABELS.md`

2.4 Finding Phrases in the Constituency Parse

Having established a source or a target of an opinion predicate with the help of the extraction rules and (normalized) dependency parsing, we need to expand sources/targets to the corresponding phrases in a constituency parse. The dependency parser only specifies relations holding between words (i.e. *heads* of phrases). For this expansion, we use a simple heuristics which applies for both opinion sources and opinion targets. Figure 5 illustrates it for opinion sources. It identifies the lowest common ancestor for the opinion verb (i.e. *kritisiert*) and the head of its source (i.e. *Polizei*). Then, we choose as the phrase the node directly dominated by the lowest common ancestor and dominating the head of the source (i.e. the NP *die Polizei*).[7] This heuristics is fairly reliable if both constituency and dependency parse provide a correct syntactic analysis of the pertaining sentence.

3 Translation-based System

Even though there currently do not exist any large datasets with sufficient labeled data for fine-grained sentiment analysis in German, there exist comparable resources for other languages, most notably for English. Therefore, we devised a translation-based system that tries to harness fine-grained labeled training data available in English. We chose the MPQA corpus (Wiebe et al., 2005). Due to the availability of annotation present in the MPQA corpus, the translation-based system only learns how to extract opinion sources from the MPQA corpus. In other words, that system will not detect any opinion targets. The pipeline of this system is illustrated in Figure 6.

The first step is to translate the MPQA corpus into German. This has been achieved by translating the raw text of this corpus by *Google Translate*[8]. Since the annotation of that corpus is not on the sentence level but on the phrase/word level, we need to align each word of a sentence with the

clearpage

corresponding word in the German translation. With the translation from *Google Translate*, we just obtain a sentence alignment. In order to obtain a word alignment, we employ *GIZA++* (Och and Ney, 2003).

Once a German version of the MPQA corpus has been reconstructed, two supervised learning classifiers are trained. The first is to detect subjective expressions or phrases. For that, we employ a conditional random field (Lafferty et al., 2001). As an implementation, we chose *CRF++*[9]. As a motivation, we chose a sequence-labeling algorithm because the task of detecting sentiment expressions or even (continuous) sentiment phrases is similar to other tagging problems, such as part-of-speech tagging or named-entity recognition. The feature templates for our sentiment tagger are displayed in Table 2. We use CRF++ in its standard configuration; as a labeling scheme, we used the simple IO-notation.

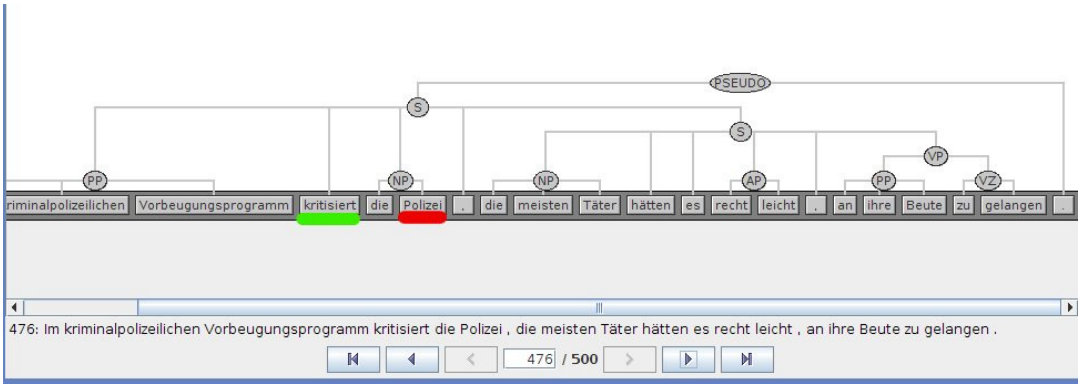
The second classifier extracts for a subjective phrase detected by the CRF the corresponding opinion source, if it exists. For this second task, a support vector machine (SVM) was chosen. As an implementation, we chose SVM^{light} (Joachims, 1999). The instance space is a set of tuples comprising candidate opinion sources (i.e. noun phrases of a sentence) and sentiment expressions/phrases (detected by the sentiment tagger). The setting is a binary classification deciding for each tuple whether the noun phrase is a genuine opinion holder of the sentiment expression/phrase, or not. Opinion sources are typically persons or groups of persons. Such entities can only be expressed by noun phrases which is why we reduce our instances to those types of constituents. SVM was chosen as a learning method since this task deals with a more complex instance space, and SVM, unlike sequence labelers, allow a fairly straightforward encoding of that instance space. The feature templates of the SVM are illustrated in Table 3.

Figure 6 indicates that a different parser (Stanford parser (Rafferty and Manning, 2008)) was used for the translation-based system compared to the rule-based system (Berkeley parser & ParZu parser). The reason for this is that those two

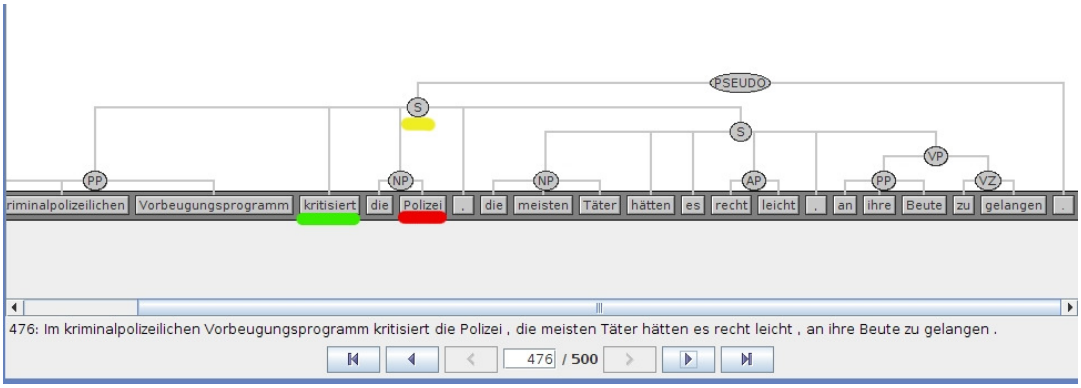
[7] Depending on the tree configuration, this node may, of course, also be a terminal node – in case the head of the source is immediately dominated by the lowest common ancestor. In such cases, the head of the source is already the constituent that we are looking for.

[8] https://translate.google.com

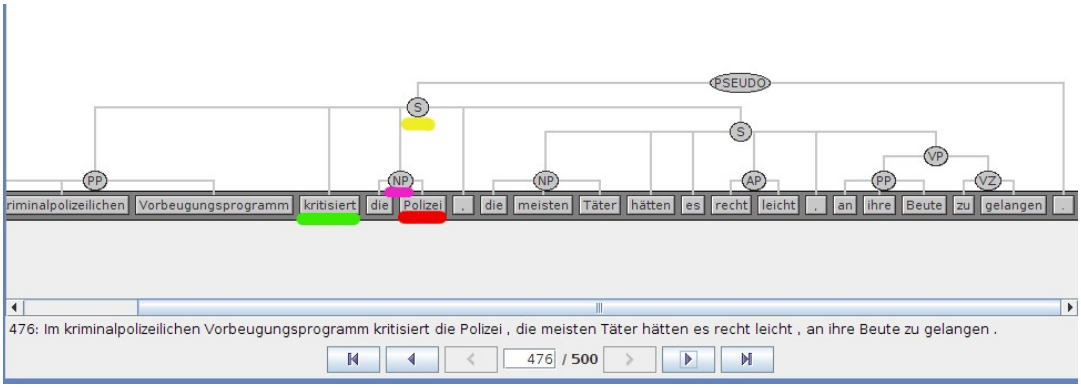
[9] https://code.google.com/p/crfpp/



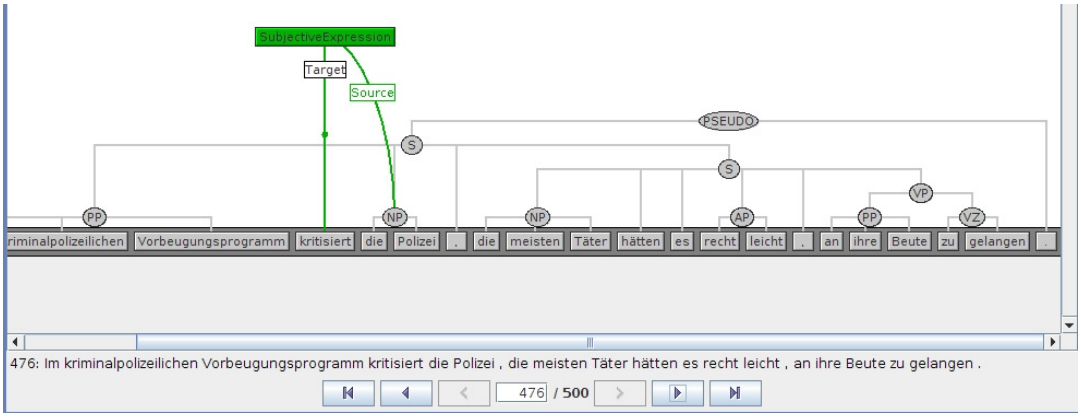
(a) start: given an opinion predicate (*kritisiert*) and the head of its source (*Polizei*)



(b) find lowest common ancestor node (*node underlined in yellow*)



(c) find direct descendant of lowest common ancestor also dominating head of source (*node underlined in violet*)



(d) final frame structure for opinion predicate and its source phrase

Figure 5: Illustration of how phrases are found for heads.

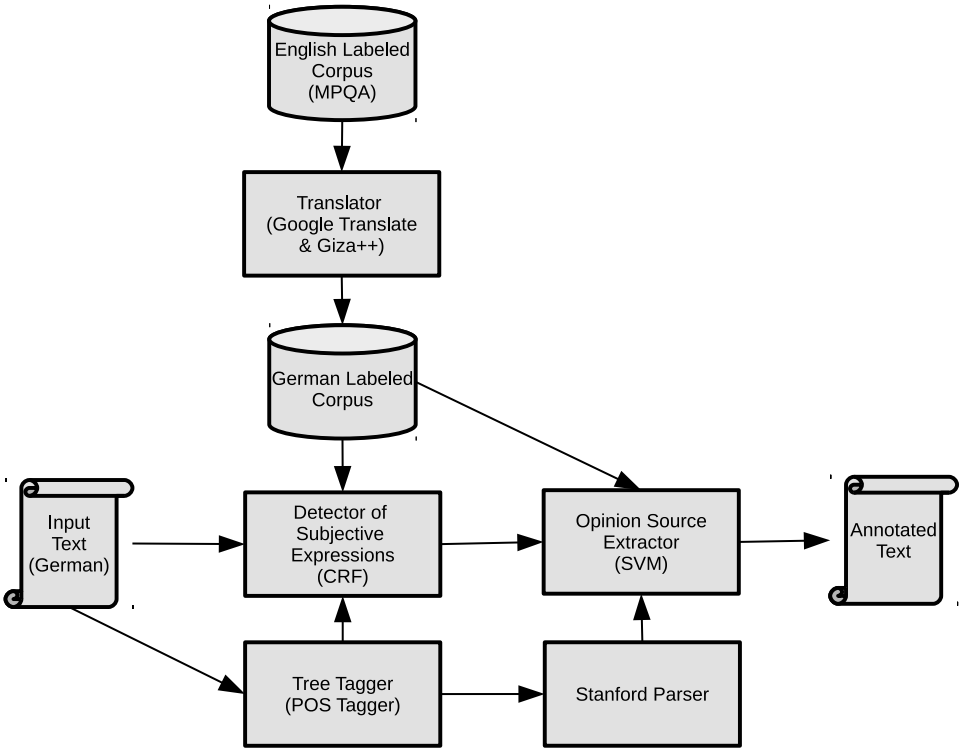


Figure 6: Processing pipeline of the translation-based system.

Type	**Feature Templates**
words	unigram features: target word and its two predecessors/successors bigrams features: bigrams of neighbouring words from unigram features
part of speech	unigram features: part-of-speech tag of target word and its two predecessors/successors bigram features: bigrams of neighbouring part-of-speech tags from unigram features bigram features: trigrams of neighbouring part-of-speech tags from unigram features
sentiment lexicon	is either of the words (window is that of the unigram features) a sentiment expression acc. to sentiment lexicon

Table 2: Feature templates employed for the CRF classifier to detect subjective expressions.

Type	**Feature Templates**
noun phrase	phrase label of noun phrase (e.g. *NP*, *MPN*, *PPER* etc.) words in phrase grammatical function if present (e.g. *SUBJ*, *OBJA* etc.)
sentiment expression	words in phrase part-of-speech tag of head of phrase
relational	distance between noun phrase and sentiment information

Table 3: Feature templates employed for the SVM classifier to detect opinion sources.

Run	Properties
Run 1	rule-based system combined sentiment lexicon dependency-parse normalization person filtering
Run 2	rule-based system combined sentiment lexicon
Run 3	rule-based system single sentiment lexicon dependency-parse normalization person filtering
Run 4	rule-based system single sentiment lexicon
Run 5	translation-based system only extracts sources

Table 4: The different properties of the different runs.

systems have been built in parallel. In particular, the superior dependency-parse normalization from the rule-based system was not implemented when that information was required for the translation-based system.

4 Experiments

In this section, we evaluate the five runs officially submitted to the shared task. Table 4 displays the different properties of the different runs. Runs 1-4 are rule-based systems, while Run 5 is a translation-based system. Runs 1 and 2 employ a large sentiment lexicon, being the concatenation of the sentiment lexicon of the PolArt system (Klenner et al., 2009) and SentiWS (Remus et al., 2010). Runs 3 and 4 are identical to Runs 1 and 2, respectively, with the exception that they only employ the sentiment lexicon of the PolArt system. Runs 1 and 3 employ normalization of the dependency parse output (Section 2.1) and person filtering for opinion sources (Section 2.3). Runs 2 and 4 neither contain normalization of the dependency parse output nor person filtering.

Table 5 displays the performance of the different configurations. **SE** evaluates the detection of subjective expressions. **Source** evaluates the detection of opinion sources, while **Source_SE** evaluates the detection of opinion sources given a correct match of subjective expression between system output and gold standard. Similarly, **Target** evaluates the detection of opinion targets, while **Target_SE** evaluates the detection of opinion targets given a correct match of subjective expression between system output and gold standard. As there is no adjudicated gold standard but 3 individual annotations provided by the different annotators for each sentence, all numbers displayed in Table 5, i.e. *precision*, *recall* and *f-score*, are the average between the system output and each of the 3 annotators' gold standards.

Table 5 shows that, on the detection of subjective expressions (SE), the combined sentiment lexicon (Runs 1 and 2) outperforms the single lexicon (Runs 3 and 4), however, the latter produces a better precision. Surprisingly, the best precision is achieved by the translation-based system (Run 5). This is most likely due to the fact that this system may be able to disambiguate subjective expressions. All rule-based systems consider each occurrence of a subjective expression in their respective sentiment lexicon as a case of a genuine sentiment.

On both the extraction of opinion sources and targets (Source and Target), the rule-based systems carrying out normalization and person filtering (Runs 1 and 3) outperform the systems without this type of processing (Runs 2 and 4). The rule-based system with the small lexicon (Run 3) outperforms its counterpart with the large lexicon on the tasks Source_SE and Target_SE since in that task, the detection of subjective expressions as such is not evaluated.

5 Conclusion

We reported on the two systems we devised for the German Shared Task on Task 1 of the German Sentiment Analysis Shared Task, the task on *Source, Subjective Expression and Target Extraction from Political Speeches (STEPS)*. The first system is a rule-based system relying on a predicate lexicon specifying extraction rules for verbs, nouns and adjectives, while the second is a translation-based system that has been obtained with the help of the MPQA corpus.

The rule-based system benefits from some linguistic processing and a large sentiment lexicon. Currently, the translation-based system is outperformed by the rule-based approach, however, there needs to be a more thorough evaluation in order to make qualified statements as to which approach is more effective for the given task. In addition, there is still plenty of space of improving either of the two approaches.

Run	Measure	SE	Source	Source_SE	Target	Target_SE
Run 1	Prec	56.83	44.35	73.16	50.40	79.57
	Rec	**36.21**	**13.73**	37.23	**19.68**	53.31
	F	**44.24**	**20.97**	49.35	**28.31**	63.85
Run 2	Prec	56.89	35.88	62.15	51.77	**80.62**
	Rec	35.97	13.06	35.64	14.87	40.58
	F	44.07	19.15	45.30	23.11	53.98
Run 3	Prec	63.42	**48.55**	**74.89**	**56.25**	79.71
	Rec	26.10	11.32	**42.46**	15.60	**58.00**
	F	36.98	18.36	**54.19**	24.43	**67.14**
Run 4	Prec	63.62	41.86	66.12	55.59	79.28
	Rec	25.80	10.98	41.68	11.74	44.19
	F	36.71	17.39	51.13	19.38	56.75
Run 5	Prec	**80.56**	47.98	58.55	*N/A*	*N/A*
	Rec	29.97	10.44	32.65	*N/A*	*N/A*
	F	43.69	17.14	41.92	*N/A*	*N/A*

Table 5: Evaluation of the different runs

Acknowledgements

This work was partially supported by the German Research Foundation (DFG) under grant WI 4204/2-1.

References

Cem Akkaya, Janyce Wiebe, and Rada Mihalcea. 2009. Subjectivity Word Sense Disambiguation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 190–199, Singapore.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of the Conference on Language Resources and Evaluation (LREC)*, pages 449–454, Genoa, Italy.

Manaal Faruqui and Sebastian Padó. 2010. Training and Evaluating a German Named Entity Recognizer with Semantic Generalization. In *Proceedings of KONVENS*, Saarbrücken, Germany.

Birgit Hamp and Helmut Feldweg. 1997. GermaNet - a Lexical-Semantic Net for German. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15, Madrid, Spain.

Thorsten Joachims. 1999. Making Large-Scale SVM Learning Practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, pages 169–184. MIT Press.

Soo-Min Kim and Eduard Hovy. 2006. Extracting Opinions, Opinion Holders, and Topics Expressed in Online News Media Text. In *Proceedings of the ACL Workshop on Sentiment and Subjectivity in Text*, pages 1–8, Sydney, Australia.

Manfred Klenner, Angela Fahrni, and Stefanos Petrakis. 2009. PolArt: A Robust Tool for Sentiment Analysis. In *Proceedings of the Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 235–238, Odense, Denmark.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the International Conference on Machine Learning (ICML)*, Williamstown, MA, USA.

George Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. 1990. Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3:235–244.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning Accurate, Compact, and Interpretable Tree Annotation. In *Proceedings of the International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics (COLING/ACL)*, pages 433–440, Sydney, Australia.

Anna Rafferty and Christopher D. Manning. 2008. Parsing Three German Treebanks: Lexicalized and Unlexicalized Baselines. In *Proceedings of the ACL Workshop on Parsing German (PaGe)*, pages 40–46, Columbus, OH, USA.

Robert Remus, Uwe Quasthoff, and Gerhard Heyer. 2010. SentiWS – a Publicly Available German-language Resource for Sentiment Analysis. In *Proceedings of the Conference on Language Resources*

and Evaluation (LREC), pages 1168–1171, Valletta, Malta.

Josef Ruppenhofer, Swapna Somasundaran, and Janyce Wiebe. 2008. Finding the Source and Targets of Subjective Expressions. In *Proceedings of the Conference on Language Resources and Evaluation (LREC)*, pages 2781–2788, Marrakesh, Morocco.

Josef Ruppenhofer, Roman Klinger, Julia Maria Struß, Jonathan Sonntag, and Michael Wiegand. 2014. IGGSA Shared Tasks on German Sentiment Analysis (GESTALT). In G. Faaßand J. Ruppenhofer, editor, *Workshop Proceedings of the 12th Edition of the KONVENS Conference*, Hildesheim, Germany, October. Universität Hildesheim.

Rico Sennrich, Gerold Schneider, Martin Volk, and Martin Warin. 2009. A New Hybrid Dependency Parser for German. In *Proceedings of the German Society for Computational Linguistics and Language Technology (GSCL)*, pages 115–124, Potsdam, German.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating Expressions of Opinions and Emotions in Language. *Language Resources and Evaluation*, 39(2/3):164–210.

Michael Wiegand and Dietrich Klakow. 2012. Generalization Methods for In-Domain and Cross-Domain Opinion Holder Extraction. In *Proceedings of the Conference on European Chapter of the Association for Computational Linguistics (EACL)*, pages 325–335, Avignon, France.

SentiBA: Lexicon-based Sentiment Analysis on German Product Reviews

Markus Dollmann
Heinz Nixdorf Institut
Universität Paderborn
Fürstenallee 11
33102 Paderborn
dollmann@mail.upb.de

Michaela Geierhos
Heinz Nixdorf Institut
Universität Paderborn
Fürstenallee 11
33102 Paderborn
geierhos@hni.upb.de

Abstract

In this paper, we describe our system developed for the GErman SenTiment AnaLysis shared Task (GESTALT) for participation in the Maintask 2: Subjective Phrase and Aspect Extraction from Product Reviews. We present a tool, which identifies subjective and aspect phrases in German product reviews. For the recognition of subjective phrases, we pursue a lexicon-based approach. For the extraction of aspect phrases from the reviews, we consider two possible ways: Besides the subjectivity and aspect look-up, we also implemented a method to establish which subjective phrase belongs to which aspect. The system achieves better results for the recognition of aspect phrases than for the subjective identification.

1 Introduction

The Maintask 2 aims at extracting aspects and subjective phrases and their relation in German product reviews (Ruppenhofer et al., 2014).

The system implementation for this shared task is based on previous unpublished work. The original goal was to use linguistic phenomena in order to determine the contextual polarity of subjective phrases for the sentiment classification of reviews at the document level. The implementation, called SentiBA, takes the three polarity classes positive, neutral and negative into account. It considers contextual valence shifter such as negation, intensifiers, modals, questions and a few rules for irony detection. The consideration of these contextual valence shifters had a great impact on the performance of the sentiment analysis task.

For GESTALT, we extended and improved the functionality of SentiBA by including aspect identification and by optimizing the recognition of subjective (polarity) words and phrases. Furthermore, we also implemented a mapping of subjective expressions to their target aspect phrases.

This paper is organized as follows: In Section 2, we sum up related work. In Section 3, the lexical resources are introduced. Section 4 provides a conceptual overview of our approach for this shared task. In Section 5, we present the results of our system obtained on the evaluation data and explain the different run settings, followed by a short discussion and conclusion in Section 6.

2 Related Work

Sentence or aspect-based sentiment analysis usually consists of two steps: First identify and then classify subjective expressions into positive and negative terms. For this task, only the subjectivity classification is of interest. Different methods have been developed to recognize subjective sentences. A common technique, the lexicon-based approach, uses lists of opinion words (e.g. Ding et al. (2008)). If a sentence contains one or more words of that list, it is assumed to be subjective. Another common approach uses machine learning techniques to extract subjective phrases by previously learned patterns. Our implementation is inspired by lexicon-based approaches, to match the subjective expressions in sentences more easily and to deal with linguistic phenomena such as valence shifters.

Valence shifters (Polanyi und Zaenen, 2004) are words and phrases that can shift or change semantic orientation. Although we ignore the semantic orientation of words and phrases for this task, we have to consider some of these valence shifters, too. Since valence shifters have an impact on subjective expressions, they should be stored together.

We identified two rules to find additional subjective expressions which are not covered by the sentiment lexicon. One of these rules introduced by Hatzivassiloglou und McKeown (1997) deals with the conjunction "and". It says that conjoined adjectives usually have the same orientation. In the sentence "This car is beautiful and spacious" where "beautiful" is known to be subjective, it can be inferred that "spacious" is also subjective. Further if "beautiful" is known to be positive, "spacious" is very likely to be also positive, because people usually express the same sentiment on both sides of a conjunction (Liu, 2012). A similar rule is about the connective "but" which is similar to the rule explained above, but has a contrary impact on the polarity of the words (Hatzivassiloglou und McKeown, 1997).

Hu und Liu (2004) present a frequency-based approach to identify aspect phrases. Nouns that are frequently used are likely to be true aspects (called frequent aspects). When different reviewers tell different (irrelevant) stories, the words used to discuss the product aspects/features converge. These words are the main aspects.

3 Resources

To identify subjective expressions, we used the sentiment lexicon SentiWS, which contains 1,650 positive and 1,818 negative word lemmas, which sum up to 15,649 positive and 15,632 negative word forms incl. their inflections (Remus et al., 2010). We also used a list of negation words and intensifiers, which were optained from the German version of SentiStrength[1].

The USAGE data set serves as training data for this shared task (Klinger und Cimiano, 2014). The data set contains annotations for more than 600 German Amazon reviews covering six different domains: Coffee machines, cutlery sets, microwaves, toasters, trash cans, vacuum cleaners and washing machines. We divided the training set into two parts. The coffee machine reviews were used to test our system. The other reviews were used to generate blacklists of subjective and aspect phrases, by counting for all expressions, how often they were correctly or incorrectly identified (see Sections 4.2 and 4.3).

We also created a subjectivity lexicon from the annotated training data provided for this main-task (except the coffee machine reviews). In the following we will call this lexicon the USAGE lexicon. We extracted all subjective words and phrases from the training data, counted the number of occurrence for each expression and created a frequency list. We tested the USAGE lexicon in conjunction with SentiWS on the coffee machine reviews and achieved better results than with SentiWS alone. Due to misidentifications in different domains, we decided to manually delete domain-dependent expressions, by the estimation of the authors. We received a list of subjective words and phrases that is domain independent and contains typical expressions used in product reviews, like "5-stars" or "strong buy recommendation". Due to these adaptations, we achieved even better results in our tests. The created USAGE lexicon contains 13 subjective words and 267 subjective phrases.

4 Implementation

In this section, we present our implementation design. Figure 1 gives an overview of the sequential steps and the required resources. These steps will be described in this section (see Sections 4.1-4.5). First, SentiBA preprocesses each product review. Subsequently the tool identifies subjective and aspect phrases. Then SentiBA indicates corresponding subjective phrases for each aspect phrase. Finally, all collected information is stored in a structured format.

4.1 Preprocessing

Before identifying subjective and aspect phrases, we preprocess each review by means of the Apache OpenNLP toolkit[2].

[1] http://www.ofai.at/research/interact/resources/SentiStrength_DE/download_form.html

[2] https://opennlp.apache.org

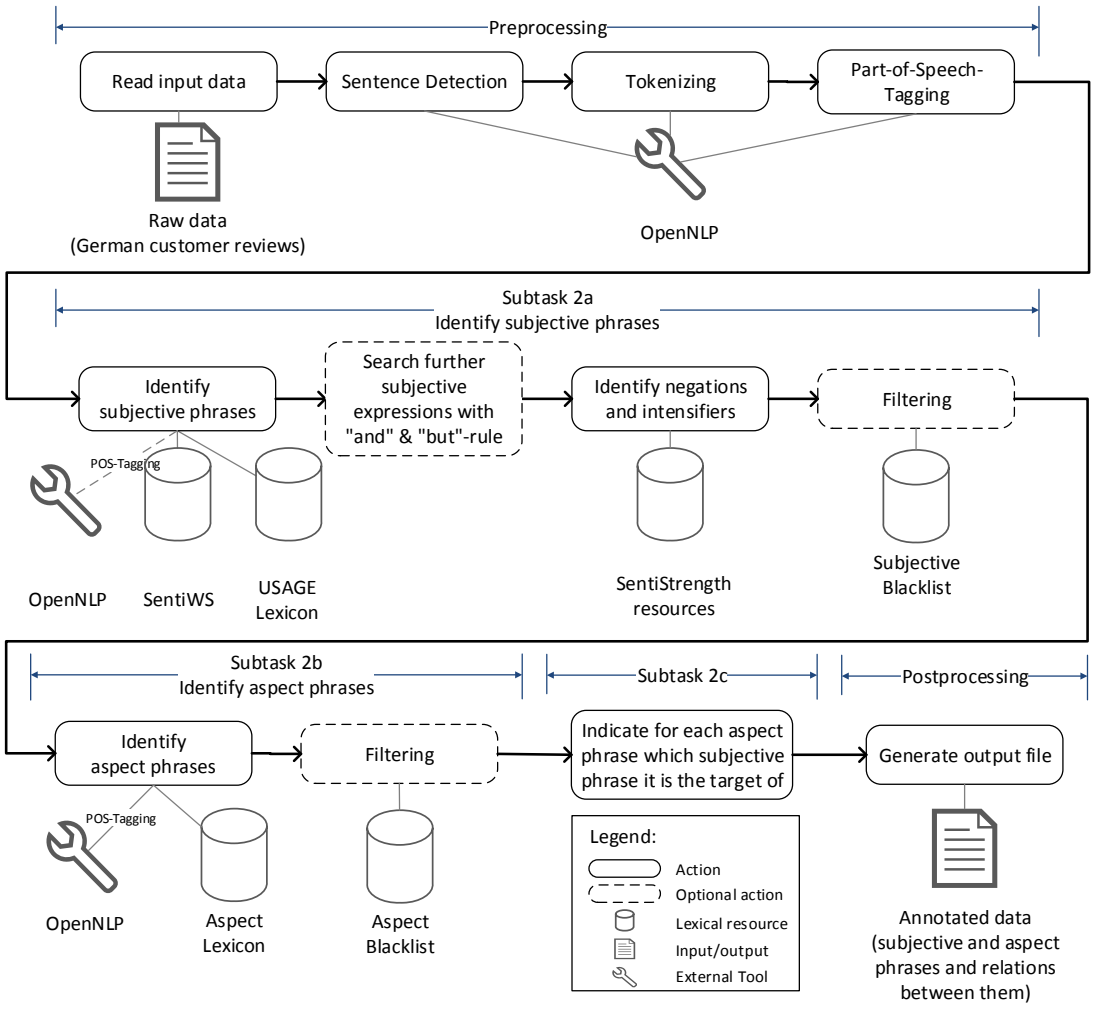


Figure 1: System overview: Steps and resource usage

We used the Sentence Detector (trained on TIGER data) from OpenNLP to split the reviews in single sentences. After that, they were tokenized by the OpenNLP Tokenizer (trained on TIGER corpus). The data structure allows us to add individual tags to every token. That way, we label tokens as subjective, aspect, negation, intensifier or any other predefined tag using the OpenNLP POS-Tagger (maxent model trained on TIGER corpus).

4.2 Subtask 2a: Identify subjective phrases

As already mentioned, we extended SentiBA by adding the sentiment lexicon SentiWS to process German reviews. We also improved the identification of subjective (polarity) words and phrases in different ways, independently from the research goal of our previous work.

To identify subjective words, SentiBA looks up every word of a review in the sentiment lexicon SentiWS. If the word exists in SentiWS, it will be annotated as subjective. When POS-Tagging is enabled, the word is only labeled as subjective if also the POS tag of the word in the review is equal to its POS tag in the lexicon. Additionally SentiBA also checks every word and phrase, in the USAGE lexicon. In this case POS tags are not considered any more.

To extend the recognized subjective words to subjective phrases, we identify negation words and intensifiers by a single token comparison with a list of negation words or intensifiers. In this

case, we add a specific tag to these words. Since we are interested in subjective phrases so far and not in the polarity of these phrases, a further processing is not necessary. In the postprocessing step (see Section 4.5) these identified negations and intensifiers will be combined with the subjective words to become phrases.

We also detect additional subjective words (which are not included in SentiWS) by using patterns with the conjunctions "and" (in German: „und") and the connective "but" (in German: „aber"). If a sentence contains the word „und" or „aber", SentiBA searches in the left and right context of the target word within a given window. If an already identified subjective word is found, SentiBA looks in the other direction of the sentence, for a given distance from the words „und" or „aber" for an unidentified adjective. In our tests, the best performance was achieved by a word distance of one, which means that the adjective and the already identified word are directly next to the word „und" or „aber". If SentiBA locates an adjective, it will label it as a subjective word. To filter common misidentified subjective expressions, we created a blacklist. To generate this blacklist, we counted for all identified subjective words and phrases from the training data (except the coffee machine reviews) how often they were correctly or incorrectly identified. Table 1 shows the most frequent misidentified subjective expressions together with their corresponding frequency of being (in)correctly identified.

Aspect	Translation	#Incorrect	#Correct
leider	sadly	55	0
gut	good	36	57
einfach	easily	28	19
alten	old	22	0
schnell	fast	22	20
alte	old	20	0
kleine	small	18	0
neue	new	18	0
genau	exactly	16	0
wieder kaufen	buy again	15	0

Table 1: 10 most frequent misidentified subjective words and phrases

4.3 Subtask 2b: Identify aspect phrases

We implemented two different approaches to identify aspect phrases in product reviews: A frequency-based approach and a naive approach, which nevertheless achieves better results.

Frequency-based approach

One approach was to identify aspect phrases through an aspect lexicon, which contains the most frequent candidates for aspect phrases from product reviews for the specific domain. We identified potential aspects by noun POS tags. The 10 most frequent potential aspects for the domain "coffee machine" are given in Table 2. We generated a frequency list for all potential aspect expressions. To identify aspects, we look up each word or phrase in that aspect lexicon, under the assumption that a specific threshold is exceeded. Surprisingly, starting by a threshold of one, the higher the threshold the lower the F-Score for the aspect identification. While the precision increases with a higher threshold, the recall drops very quickly. Our second approach achieved considerable better results.

Aspect	Translation	Frequency
Kaffee	coffee	90
Maschine	machine	71
Kaffeemaschine	coffee machine	67
Kanne	pot	35
Wasser	water	20
Preis	price	13
Gerät	device	12
Thermoskanne	thermos	11
Tassen	mugs	11

Table 2: 10 most frequent aspect candidates for coffee machines

Does each noun describe an aspect?

The more satisfying approach is also based on the POS tag for nouns. Instead of the frequency-based approach, SentiBA now assumes that every noun in the product review represents an aspect. Just like in the subjectivity identification, we created a blacklist to filter common misidentified expressions. To generate this blacklist, we counted for all identified nouns (and noun phrases) from the training data (except the coffee machine reviews) how often they were correctly or incorrectly identified. Table 3 shows the most frequent misidentified aspects together with their corresponding frequency of being (in)correctly identified. This very simple approach achieves remarkably better results in our tests on the coffee machine reviews.

Aspect	Translation	#Incorrect	#Correct
Zeit	time	36	24
Jahre	years	27	0
Jahr	year	26	0
Gebrauch	use	23	4
Für	for	22	0
Jahren	years	22	0
Probleme	problems	21	0
Fazit	conclusion	21	0
Problem	problem	18	0
Tag	day	17	0

Table 3: 10 most frequent misidentified aspects

4.4 Subtask 2c: Indicate for each aspect phrase which subjective phrase it is the target of

We applied a quite simple approach to indicate corresponding subjective phrases for each aspect phrase. SentiBA calculates for each identified aspect phrase from Subtask 2b the token distance to every identified subjective phrase, which is in the same sentence as the aspect phrase. The subjective phrase with the shortest distance to the aspect phrase will be taken as the subjective expression for that aspect phrase.

This approach can easily be extended in future by adding multiple subjective phrases to aspects, e.g. if multiple subjective phrases in the same sentence are connected by words like “and” or “but”. Moreover, coreference resolution is not considered in this approach. A possible attempt could be to search backward for the next aspect phrase and match the coreference word with this aspect.

4.5 Postprocessing

In the postprocessing step SentiBA stores all previously collected information into two output files: One file for the identified subjective and aspect phrases and one file for the relations between them.

SentiBA saves every word of the input review, which was tagged as subjective in the output file. Therefore SentiBA links the neighboring subjective words to phrases and also adds neighboring negations and intensifiers to these words or phrases. It is done in a similar way for the identified aspect words, while neighboring aspect words are saved as an aspect phrase. Additionally the identified relations from Subtask 2c are stored in the relation file.

5 Results

SentiBA was tested with different settings. Because of the poor results during our own tests, we decided to drop the frequency-based aspect identification approach and only pursued the approach presupposing each noun as an aspect.

We devided our evaluation runs as shown in Table 4. In three of five runs we used the subjective

Run	Blacklists	POS-Tagging	“and”& “but”-rule
1	√	X	X
2	√	X	√
3	√	√	√
4	X	X	X
5	X	√	X

Table 4: Settings for the different runs

	Precision	Recall	F_1
Run 1			
Subtask 2a	**0.527**	0.312	0.392
Subtask 2b	**0.555**	0.622	**0.587**
Subtask 2c	**0.126**	0.138	**0.132**
Run 2			
Subtask 2a	0.516	0.320	0.395
Subtask 2b	**0.555**	0.622	**0.587**
Subtask 2c	0.124	0.138	0.131
Run 3			
Subtask 2a	0.503	0.260	0.342
Subtask 2b	0.530	0.614	0.569
Subtask 2c	0.118	0.117	0.118
Run 4			
Subtask 2a	0.443	0.359	0.396
Subtask 2b	0.477	**0.650**	0.550
Subtask 2c	0.095	**0.148**	0.116
Run 5			
Subtask 2a	0.432	**0.367**	**0.397**
Subtask 2b	0.477	**0.650**	0.550
Subtask 2c	0.092	0.143	0.112

Table 5: Results from the different runs on the test data

and aspect blacklists to filter common misidentified subjective and aspect expressions. Although these blacklists had a positive influence during our tests on the coffee machines, we decided to also perform runs without these blacklists, if the main aspect or subjective words and phrases of the new category are part of these blacklists. We also decided to have runs with and without POS-Tagging. POS-Tagging helps to identify different word senses, but also decreases the number of recognitions in the lexicon. The last difference in the runs is the application of rules to identify new subjective words by usage of the conjunction "and" and the connective "but".

We decided to have runs in- and excluding these rules, in order to examine whether new subjective words can be identified with this method. But the error rate should not be underestimated.

The results from the different runs on the test data are given in Table 5. The best results for identifying subjective phrases (see F-Score in Subtask 2a) were achieved by run no. 5, where the subjective blacklist was not used, POS-Tagging was enabled and the both conjunction-rules were disabled. The usage of POS-Tagging improves the recall, but decreases the precision (compare with run no. 4). The usage of the subjective blacklist increases the precision remarkably, but decreases the recall seriously.

The best results for identifying aspect phrases (see F-Score in Subtask 2b) were achieved by the runs no. 1 and no. 2, when the aspect blacklist was used and POS-Tagging was disabled. The usage of the "and" & "but"-rules had no impact on the aspect identification.

The results for the matching of aspect phrases to subjective phrases depend on the results of Subtask 2a and 2b. The best result was delivered by run no. 1, where also the aspect identification achieved the best result.

In comparison to our own evaluation on the coffee machine reviews (see Table 6) the results on the test data are poorer. The best F-Score reached on the test data by identifying subjective phrases is 0.397, on the coffee machine reviews the score is 0.453. For identifying aspect phrases, the best F-Score on the test data is 0.587, while on the coffee machine reviews it is 0.634.

	Run 1	Run 2	Run 3	Run 4	Run 5
Subtask 2a	**0.453**	0.452	0.366	0.431	0.359
Subtask 2b	0.663	0.663	**0.634**	0.620	0.595
Subtask 2c	**0.199**	0.195	0.158	0.168	0.135

Table 6: F-Scores from runs on coffee machine reviews from training data (Annotator 1)

SentiBA achieves an F-Score of 0.132 on the test data for matching aspect phrases with subjective expressions, while it achieves on the coffee machine reviews a score of 0.199. This shows, that SentiBA together with the sentiment lexicon SentiWS is highly domain sensitive.

6 Conclusion and Future Work

We presented a system for subjective phrase and aspect extraction from product reviews. We pursued a lexicon-based approach using SentiWS and a newly created and manually edited subjective lexicon from the training data. To identify aspect phrases, we implemented two approaches: A frequency-based approach, which identifies aspect phrases through an aspect lexicon that contains the most frequent candidates for aspect phrases and an even more satisfying approach based only on the noun POS tag, where our system assumes that every noun in the product review represents an aspect. We also conducted a simple matching method that assigns each aspect phrase to its corresponding subjective phrase. While the system achieves satisfactory results in the recognition of aspect phrases, the subjective identification and especially the matching should be improved in further work. The comparison between the results from the test data and the results from an excluded part of the training data showed that our implementation is highly domain sensitive. Moreover it shows that the different run settings in various domains have varying results. The frequent nouns approach for identifying aspect phrases gave poor results on the test data; so it was not used in the test runs. In future work, this approach could be improved by searching frequent nouns on a bigger training corpus or by searching for more reviews from the same domain in the Internet. The matching of aspect and subjective phrases could be improved by applying coreference resolution and by further research for better rules to indicate which subjective phrase belongs to which aspect phrase.

References

Xiaowen Ding, Bing Liu, und Philip S. Yu. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, S. 231–240, New York, NY, USA. ACM.

Vasileios Hatzivassiloglou und Kathleen R. McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, ACL '98, S. 174–181, Stroudsburg, PA, USA. Association for Computational Linguistics.

Minqing Hu und Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, S. 168–177, New York, NY, USA. ACM.

Roman Klinger und Philipp Cimiano. 2014. The usage review corpus for fine grained multi lingual opinion analysis. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, und Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

B. Liu. 2012. *Sentiment Analysis and Opinion Mining*. Synthesis digital library of engineering and computer science. Morgan & Claypool.

Livia Polanyi und Annie Zaenen. 2004. Contextual valence shifters. In *Working Notes — Exploring Attitude and Affect in Text: Theories and Applications (AAAI Spring Symposium Series)*.

R. Remus, U. Quasthoff, und G. Heyer. 2010. Sentiws – a publicly available german-language resource for sentiment analysis. In *Proceedings of the 7th International Language Resources and Evaluation (LREC'10)*, S. 1168–1171.

Josef Ruppenhofer, Roman Klinger, Julia Maria Struß, Jonathan Sonntag, und Michael Wiegand. 2014. Iggsa shared tasks on german sentiment analysis (gestalt). In Gertrud Faaßund Josef Ruppenhofer, editors, *Workshop Proceedings of the 12th Edition of the KONVENS Conference*, Hildesheim, Germany, October. Universität Hildesheim.

University of Hildesheim

Institute of Information Science and Natural Language Processing
Marienburger Platz 22
D-31141 Hildesheim

https://www.uni-hildesheim.de/iwist/

https://www.uni-hildesheim.de/konvens2014/